VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra kybernetiky a biomedicínského inženýrství

Systém pro syntézu mluvených hlásek pro tvorbu umělého mluveného signálu

Synthesis Speech Vovels System for Artificial Speech Signal Generation

Ostrava, 2018 Bc. Jan Matyáš

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra kybernetiky a biomedicínského inženýrství

Zadání diplomové práce

Student: **Bc. Jan Matyáš**

Studijní program: N2649 Elektrotechnika

Studijní obor: 3901T009 Biomedicínské inženýrství

Téma: Systém pro syntézu mluvených hlásek pro tvorbu umělého mluveného signálu
Synthesis Speech Vovels System for Artificial Speech Signal Generation

Jazyk vypracování: čeština

Zásady pro vypracování:

1. Rozbor problematiky syntézy mluveného signálu s popisem metod.
2. Návrh systému pro syntézu umělého mluveného signálu.
3. Realizace systému pro konkatenační syntézu mluveného signálu, kde je implementována minimálně jedna metoda, založená na modelu řečového traktu prostřednictvím lineární predikce nebo z kepstra.
4. Extrakce řečových jednotek s využitím manuálně segmentovaných difonů pro srozumitelnost generovaného zvukového signálu ve formě jednoduchých větných spojení.
5. Vizualizace a srovnání naměřených výsledků s teoretickými předpoklady.
6. Zhodnocení dosažených výsledků závěrečné práce.

Seznam doporučené odborné literatury:

[1] BROUGHTON, S. Allen a Kurt M. BRYAN. *Discrete Fourier analysis and wavelets: applications to signal and image processing.* Hoboken, N.J.: Wiley, c2009, xv, 337 p. ISBN 978-0-470-29466-6.
[2] CLARK, Alexander, Chris FOX a Shalom LAPPIN. *The Handbook of Computational Linguistics and Natural Language Processing.* Malden, MA: Wiley-Blackwell, 2010, xxii, 775 p. ISBN 978-1-4051-5581-6.
[3] HUANG, Xuedong, Alex ACERO a Hsiao-Wuen HON. *Spoken language processing: a guide to theory, algoritm, and system development.* New Jersey: Prentice-Hall, 2001. 980 s. ISBN 0-13-022616-5.
[4] PSUTKA, Josef. *Komunikace s počítačem mluvenou řečí.* Praha: Academia Praha, 1995. ISBN 80-200-0203-0.
[5] GOLD, Ben, Nelson MORGAN a Dan ELLIS. *Speech and Audio Signal Processing.* Hoboken, NJ.: John Wiley & Sons, 2011. ISBN 978-0-470-19536-9.
[6] PSUTKA, Josef, et al. *Mluvíme s počítačem česky.* Praha: Academia Praha, 2006. ISBN-80-200-1309-1.
[7] NOUZA, Jan. *Počítačové zpracování řeči:cíle, problémy, metody a aplikace.* Liberec: Technická univerzita v Liberci, 2001. ISBN 978-8070835517.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. Zdeněk Macháček, Ph.D.**

Konzultant diplomové práce: doc. Ing. Martin Augustynek, Ph.D.

Datum zadání: 01.09.2016

Datum odevzdání: 30.04.2018

doc. Ing. Jiří Koziorek, Ph.D.
vedoucí katedry

prof. Ing. Pavel Brandštetter, CSc.
děkan fakulty

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením Ing. Zděnka Macháčka, Ph.D a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v přiloženém seznamu.

V Ostravě dne 27. 4. 2018

podpis

Poděkování

Rád bych touto cestou vyjádřil poděkování Ing. Zdeňkovi Macháčkovi, Ph.D za cenné rady a připomínky, vedoucí k zakončení této práce.

Abstrakt

MATYÁŠ, J.: Systém pro syntézu mluvených hlásek pro tvorbu umělého mluveného signálu. Ostrava 2018. 62 stran. Diplomová práce. VŠB Technická univerzita Ostrava. Vedoucí práce: Ing. Zdeněk Macháček, Ph.D.

Cílem této diplomové práce je návrh a realizace systému pro konkatenační syntézu mluveného signálu v češtině. Tento systém je založený na modelu řečového traktu prostřednictvím lineární predikce a převádí psaný text na umělý řečový signál. Z předem nahraných úseků řeči jsou vymezeny jednotky a pomocí algoritmů zakódovány v inventářích. Výsledná řeč je tvořena na základě řetězení rekonstruovaných segmentů jednou ze zvolených tří metod. Syntetizér je nakonec zhodnocen poslechovými testy MRT a MOS.

Klíčová slova

lineární predikce, LPC, RELP, RPE-LPC, syntéza řeči, TTS

Abstract

MATYÁŠ, J.: Synthesis Speech Vovels System for Artificial Speech Signal Generation. Ostrava 2018. 62 pages. Master's thesis. VŠB Technical university Ostrava. Thesis advisor: Ing. Zdeněk Macháček, Ph.D.

The aim of this Master's thesis is to design and implement a concatenative speech synthesizer in Czech language. This system is built on linear predictive coding based vocal tract model and converts written text to a syntetic speech signal. Units are segmented from previously recorded prompts, encoded by algorithms and stored in inventories. The resulting speech is created by concatenation of reconstructed segments by one of the three selected methods. Lastly, the synthesizer is evaluated by MRT and MOS tests.

Key Words

linear prediction, LPC, RELP, RPE-LPC, speech synthesis, TTS

Obsah

Seznam použitých zkratek a symbolů

ACF	autokorelační funkce	
AR	autoregresní model	
B	počet kvantizačních bitů	
CPD	kepstrální detektor znělosti	
C_L	omezující úroveň	
DP	dolní propust	
F_0	základní tón řeči	[Hz]
F_{vz}	základní tón řeči	[Hz]
F_m	mezní frekvence	[Hz]
G	intenzita řeči	
HMM	skryté Markovovy modely	
H(z)	přenosová funkce systému	
IPA	mezinárodní fonetická abeceda	
IPK1, IPK2	maximální absolutní hodnota mikrosegmentů	
L	počet vzorků mezi hlasivkovými pulsy	
LAR	koeficienty filtru (log area ratio)	
LP, LPC	lineární predikční kódování	
MACF	modifikovaná autokorelační funkce	
MATLAB	vývojové prostředí (matrix laboratory)	
MPE-LPC	LPC buzené pulsně vyjádřeným reziduálním signálem	
N	velikost segmentu/okénka ve vzorcích	
OS	operační systém	
PARCOR	reflexní koeficienty	
RELP	lineární predikce buzená reziduálním signálem	
R(m)	krátkodobá autokorelační funkce	
RPE-LPC	buzení pulsně vyjádřeným reziduálním signálem	
SAMPA	fonetická abeceda pro počítač	
SAPI	aplikační programové rozhraní pro řeč	
s_n	sekvence vzorků diskretizovaného signálu	
STE	krátkodobá energie	
T_0	perioda základního tónu řeči	[s]
TTS	systém univerzální syntézy řeči	
ZCR	počet průchodů nulou	

Seznam obrázků

Seznam tabulek

1. Úvod

Syntéza řeči je počítačově řízený převod zadaného textu do slyšitelné a srozumitelné umělé řeči. Tyto systémy mají hojné využití v mnoha oblastech, jako například předčítací programy pro zrakově postižené lidi, výuku jazyků a v dalších aplikacích, ve kterých je dostupný text v digitální podobě a je potřeba řečový doprovod. Lze je také využít jako systémy kódování řeči a dosáhnout velmi dobré komprese dat. Systémy TTS (Text-to-speech) nabízí velkou flexibilitu v situacích, kdy se vstupní text často mění a není tedy možné nahrání všech možných zpráv.

Generování syntetického hlasu, který přesně imituje lidskou řeč není ovšem triviální proces, jelikož je obecně požadováno hodně informací o daném jazyku, kontextu a hluboké porozumění sémantiky obsahu textu.

Tato práce se zabývá návrhem a realizací vlastního konkatenačního systému pro syntézu řeči s využitím techniky lineární predikce pro kódování segmentů a použití difónů jako řečových jednotek.

Druhá kapitola velmi stručně popisuje mechanismy tří základních druhů produkce řeči hlasovým traktem. Úkolem syntetizéru je matematicky modelovat tyto procesy.

Ve třetí kapitole je popsána analýza řečového signálu. Jsou zde ukázány základní procesy digitalizace řeči, dále proces segmentace řeči na kvazistacionární úseky a nakonec je vysvětlen pojem základní frekvence hlasivkového tónu a jsou popsány algoritmy pro jeho odhad. Celý proces vybrané metody MACF je vysvětlen krok po kroku.

Čtvrtá kapitola popisuje druhy syntézy řeči se zaměřením na konkatenační syntézu. Následuje výběr typu řečových jednotek a popis inventáře, ve kterém se ukládají.

Pátá kapitola se zabývá technikou lineární predikce na které celý syntetizér staví a je popsáno tvoření umělého signálu modelem LP pomocí tří metod s různými typy buzení. První metoda má zjednodušený budící signál, který modeluje znělé části řeči posloupností impulsů se zjištěnou periodou kmitání hlasivek a neznělé části řeči bílým šumem. Metoda RELP využívá pro buzení filtru reziduální signál (chyba predikce) a metoda RPE-LPC navazuje na metodu RELP s tím, že reziduální signál reprezentuje určitým počtem rovnoměrně rozmístěných impulsů.

V šesté kapitole je uvedeno několik příkladů nejpoužívanějších dostupných TTS systémů s českými hlasy.

Sedmá kapitola popisuje celý proces návrhu a realizace vlastního systému pro syntézu řeči se všemi náležitostmi. Jsou zde popsány algoritmy programového řešení, tvoření umělé řeči od nahrání řečového korpusu, jeho analýzy, uložení do slovníku, zpětné syntézy a zřetězení segmentů do výsledného signálu. Dále obsahuje popis vytvořených interaktivních grafických prostředí.

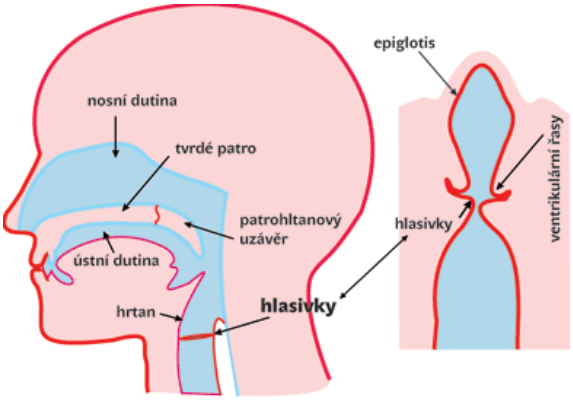
Poslední kapitola je určená pro vyhodnocení výsledků a porovnání metod dle subjektivních poslechových testů MRT a MOS.

2. Vznik řečového signálu

Řeč vzniká když vzduch z plic projde přes hlasivkovou štěrbinu, skrze hrdlo a nakonec vychází ústy. Řečový signál může být tvořen třemi typy excitace.

- **Znělá excitace.** Tlak vzduchu z plic prochází přes hrtan a periodicky otevírá a zavírá hlasivkovou štěrbinu (glottis) a tím generuje periodický sled impulsů se základní frekvencí F_0. Tato frekvence se u každého člověka liší, neboť záleží na anatomii hrtanu.
- **Neznělá excitace.** Při otevřených hlasivkách prochází vzduch do úzké komory v hrdlu nebo v ústech. Vzniká turbulence, která generuje šumový signál. Spektrální obálka tohoto šumu závisí na místě zúžení. V takovém signálu není pozorovaná periodicita jako u znělé excitace.
- **Přechodná excitace.** Dočasné sevření úst, či hrdla navýší tlak vzduchu a při náhlém otevření se tento tlak rapidně sníží.

Ve většině případů se výsledný zvuk skládá z kombinace všech tří typů excitace. Spektrální obálka závisí na tvaru hlasového traktu (trubice, tvořená hrdlem, jazykem, zuby a rty). Nedílná součást procesu vytváření srozumitelné řeči je také oddělení znělé a neznělé řeči tichem. V tomto případě se do hlasového traktu nedodává žádný excitační signál.



Obrázek 2.1: Hlasové ústrojí [1].

3. Analýza řečového signálu

Veškeré zpracování řečového signálu probíhá v jeho digitální formě. Je tedy nutné převést analogový signál řečníka na číslicový procesem zvaným *digitalizace*.

3.1 Digitalizace

Digitalizace, nebo-li převod spojitého (analogového) signálu na číslicový (digitální) a skládá ze tří kroků: vzorkování, kvantizace a kódování.

3.1.1 Vzorkování

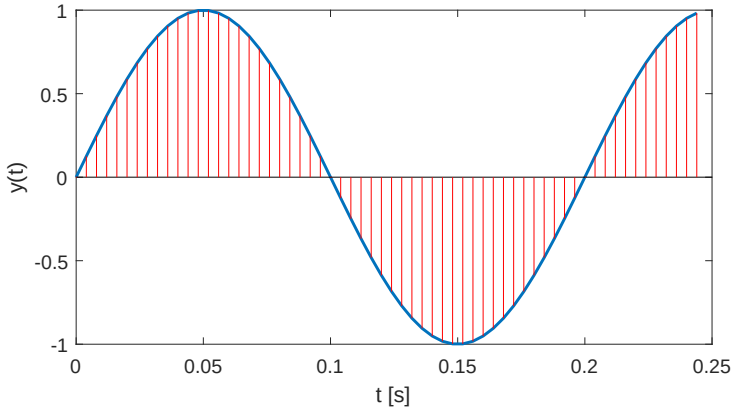
Jedná se o proces zaznamenávání hodnot analogového signálu v určitých časových momentech. Vzorkovací frekvence F_{vz} [Hz] udává počet vzorků za sekundu. Časový interval mezi vzorky se nazývá vzorkovací perioda T_s.

Typicky vzorek s_0 se bere v čase t = 0 analogového signálu. Je tedy zřejmé, že vzorek s_1 je brán v čase t = T_s, přesně jednu vzorkovací periodu později. Sekvenci vzorků diskretizovaného signálu s(t) lze tedy vyjádřit jako:

$$s_n = s(nT). \tag{3.1}$$

Při volení vzorkovací frekvence je nutné dbát na Nyquistův-Shannonův teorém, který praví, že přesná rekonstrukce frekvenčně omezeného signálu je možná pouze pokud je vzorkovací frekvence vyšší než dvojnásobek nejvyšší harmonické složky vzorkovaného signálu [8].

$$F_{vz} \geq 2F_m. \tag{3.2}$$



Obrázek 3.1: Vzorkování signálu (F_{vz} = 250 Hz).

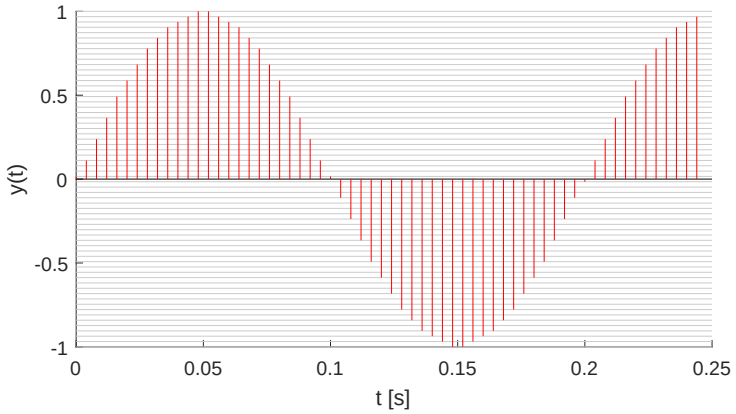
Na obrázku 3.1 je zobrazen generovaný sinusový průběh signálu o frekvenci 5 Hz, který představuje analogový vstupní signál. Vzorkovací frekvence F_{vz} je pro lepší znázornění nastavena na menší než používanou hodnotu v řečových systémech (250 Hz). Systém vezme po určitém časovém intervalu ($\tau = \frac{1}{F_{vz}} = \frac{1}{250} = 4$ ms) jeden vzorek z průběhu a reprezentuje tedy vstupní spojitý signál konečným počtem hodnot.

Aby byl splněn Nyquistův teorém na skutečných řečových signálech je nutno vzorkovat minimálně dvojnásobkem nejvyšších frekvencí v řeči (4 kHz) a tedy $F_{vz} = 2 \cdot 4000 = 8$ kHz [8].

3.1.2 Kvantování a kódování

Sekvence hodnot s_n z rovnice 3.1 a obrázek 3.1 zatím není digitální signál, protože jeho vzorky můžou nabýt jakýkoliv hodnot ze spojitého rozsahu.

Kvantizací je každý vzorek aproximován jednou z konečného počtu číselných hodnot 2^B (B udává počet kvantizačních bitů v binární soustavě).



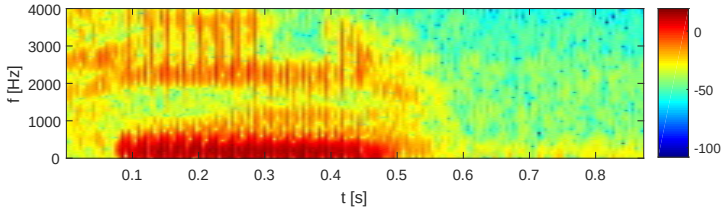
Obrázek 3.2: Kvantizace signálu (B = 6).

Obrázek 3.2 zobrazuje proces kvantizace signálu. Pro názornou ukázku byl zvolen počet kvantizačních bitů B = 6, tedy vzorky mohou být aproximovány $2^6 = 64$ hodnotami. Skutečná hodnota každého vzorku je porovnána a reprezentována jednou z řady 64 hodnot. Jedná se pouze o ilustrační příklad a samozřejmě platí, že čím více bitů, tím je přesnější reprezentace původního signálu. Rozdíl skutečné a jeho aproximované hodnoty se nazývá *kvantizační chyba* a je vhodné ji co nejvíce minimalizovat.

3.2 Spektrogramy

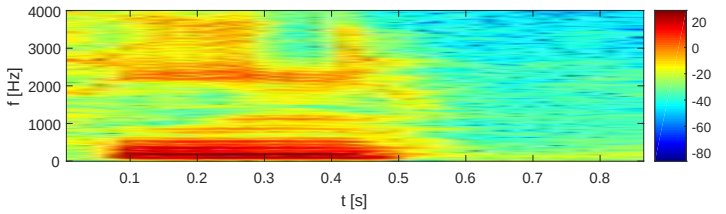
Užitečnou reprezentaci řečového signálu nabízí tzv. spektrogramy. Zobrazují vývoj krátkodobého amplitudového spektra v časové oblasti. Vrcholky ve spektru signálu lze pozorovat jako horizontální pásy odlišného zabarvení. Základní myšlenka je vypočítat krátkodobé amplitudové spektrální charakteristiky (získáme informace o amplitudě a frekvencích) v navazujících segmentech signálu. Čas je většinou zobrazen na horizontální ose a frekvence na vertikální ose. Amplituda je kódována barvou (v typické škále červená barva značí vysoké hodnoty a modrá nízké). Nejčastěji se používají dva typy spektrogramů s ohledem na jejich oblast použití: širokopásmové a úzkopásmové [4]. Liší se délkou váhového okénka při zpracování.

Širokopásmové spektrogramy používají kratší okénka (5–10 ms) a poskytují větší detaily v časové oblasti. Jejich frekvenční rozlišení je ale špatné (princip neurčitosti). Místo přesného zobrazení vlastních frekvencí signálu spíše zvýrazňují spektrální obálku a umožňují sledovat vývoj formantů (oblast lokálního maxima ve spektru tónů) v čase. Znělé segmenty řeči jsou zobrazeny jako posloupnost svislých pruhů. Širokopásmový spektrogram je vysoce využívaný nástroj pro spektrální analýzu signálů [8].



Obrázek 3.3: Širokopásmový spektrogram slova *Síň*.

Úzkopásmové spektrogramy používají delší okénka (asi 30 ms) a jsou využívány méně. Zvýrazňují jemnou spektrální strukturu a lze z nich vyčíst jednotlivé harmonické frekvence, které odpovídají frekvenci základního tónu F_0. Zobrazují se jako horizontální pruhy. Mají ale špatné časové rozlišení a nejsou proto vhodné k analýze rychle měnících se zvuků (např. plozivy). Pokud je výskyt těchto parazitních horizontálních a vertikálních pruhů v grafu nechtěný, je možné udělat kompromis a nastavit délku okénka na přibližně 2–3 násobek periody základního tónu [4].



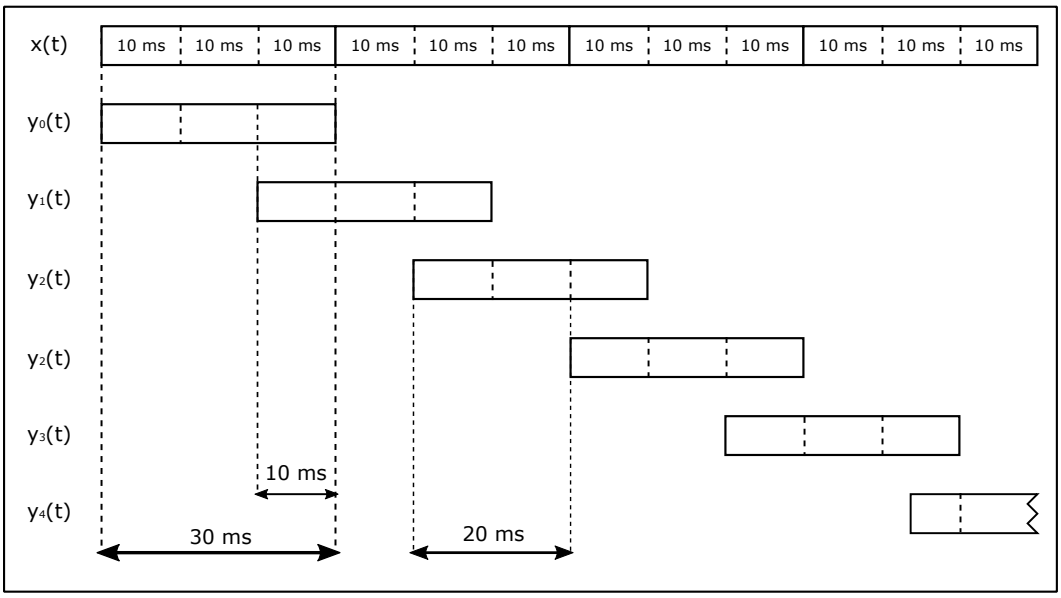
Obrázek 3.4: Úzkopásmový spektrogram slova *Síň*.

3.3 Segmentace řečového signálu

Před samotnou analýzou lidské řeči je nutno uvažovat, že se jedná o nestacionární signál. Vstupní řeč je tedy rozdělena na jednotlivé segmenty o určité velikosti, ve které lze považovat signál za téměř stacionární. Zpravidla jsou tyto segmenty dlouhé 20–30 ms. Ve standardních algoritmech jsou rozmístěny rovnoměrně po celém řečovém signálu s určitou mírou překrytí. Pro pozdější vyhlazení nespojitostí jsou segmenty překrývány o určitou velikost, např. při třetinovém překrytí a délkou segmentu 30 ms začíná nový segment každých 20 ms a je tedy překrýván o 10 ms z každé strany. Tento proces je znázorněný na obrázku 3.5.

Na každý segment je použito Hammingovo okénko. Poté se z nich extrahují všechny potřebné informace, které zahrnují: rozhodnutí, zda je rámec znělý či neznělý, intenzitu hlasu (gain), základní hlasivkovou frekvenci a koeficienty filtru.

Na rozhodnutí, zda je daný rámec znělý, či neznělý je třeba zjistit, zda má rámec dominantní frekvenci. Pokud ji má, je znělý a můžeme z něho získat i základní hlasivkovou frekvenci F_0. Pokud se v rámci nenachází dominantní frekvence, je klasifikován jako neznělý.



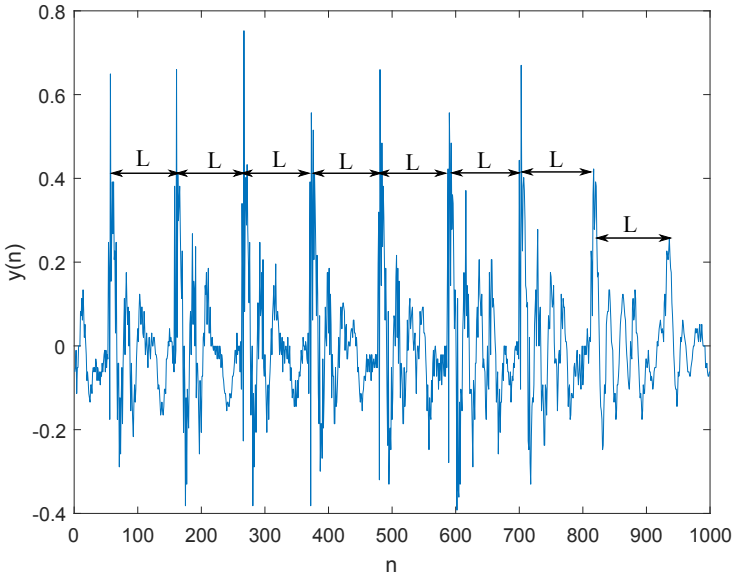
Obrázek 3.5: Segmentace řečového signálu.

3.4 Základní frekvence řeči

Základní frekvence F_0 je nejnižší frekvenční složka budícího signálu systému. Vyjadřuje kmitání hlasivek při tvoření znělých částí řeči. Tato frekvence bývá posluchačem vnímána jako výška hlasu [6]. U mužů se většinou pohybuje zhruba v rozmezí 50 Hz až 250 Hz a 120 až 500 Hz u žen [11].

Změny tónu a intonace syntetické řeči se dosahuje pomocí manipulace F_0. Tonální jazyky (např. čínské jazyky) používají změny F_0 na rozlišení významu slov. Intonace je na úrovni frází a slouží především na rozlišení typu věty (oznamovací, či tázací).

Ve skutečnosti se nejedná o konstantní hodnotu, jelikož hlasivkové pulsy nejsou ryze periodické. Objevuje se mírné kolísání amplitudy i délky periody jednotlivých pulsů.



Obrázek 3.6: Znázornění hlasivkových pulsů ve znělých hláskách.

V obrázku 3.6 lze vidět úseky, které se periodicky opakují. Počet vzorků mezi jednotlivými hlasivkovými pulsy zde reprezentujeme písmenem L (v anglické literatuře *lag*). Délka času, za kterou nastává další hlasivkový puls se nazývá perioda základního hlasivkového tónu T_0.

$$T_0 = \frac{F_{vz}}{L}, \tag{3.3}$$

kde T_0 je perioda základního tónu v sekundách, F_{vz} je vzorkovací frekvence (8 kHz) a L je počet vzorků mezi hlasivkovými pulsy.

Základní frekvenci F_0 získáme převrácenou hodnotou periody základního tónu T_0.

$$F_0 = \frac{1}{T_0}, \tag{3.4}$$

kde F_0 je základní frekvence řeči v Hz a T_0 je perioda základního tónu v sekundách.

3.5 Krátkodobé charakteristiky

Počet průchodů nulou

ZCR (Zero-crossing Rate) detektory představují jednoduchou techniku, která počítá počet průchodů nulovou referenční hladinou. Tato technika je velmi jednoduchá, ale ne příliš přesná. Při analýze velmi zašuměných signálů má špatné výsledky [2].

Tato charakteristika dokáže popisovat spektrální vlastnosti zkoumaného signálu a je ji možné využít např. pro určení začátku a konce promluvy, rozlišení znělosti hlásek a zjištění frekvence základního hlasivkového tónu F_0 [12].

Krátkodobou funkci středního počtu průchodu nulou lze definovat jako

$$ZCR = \sum_{k=-\infty}^{\infty} |sgn[s(k)] - sgn[s(k-1)]|\, w(n-k), \tag{3.5}$$

kde

$$sgn[s(k)] = \begin{cases} 1 & \text{pro } s(k) \geq 0 \\ -1 & \text{pro } s(k) < 0 \end{cases}, \tag{3.6}$$

a *w(n)* je okénko (pravoúhlé).

Krátkodobá energie

Charakteristika krátkodobé energie signálu *STE* (Short Term Energy) obsahuje informaci o průměrné hodnotě energie v segmentu.

Lze ji definovat jako

$$STE = \sum_{k=-\infty}^{\infty} [s(k)w(n-k)]^2, \tag{3.7}$$

kde *w(n)* je vybraný typ okénka a *s(k)* je jeden vzorek signálu v čase k.

Tuto charakteristiku lze využít např. v kombinaci se ZCR pro oddělení znělých a neznělých segmentů nebo dělení regionu ticha od aktivní řeči [11].

3.6 Detektory znělosti a odhad základní frekvence

Detektory znělosti jsou algoritmy pro určení, zda je analyzovaný řečový segment znělý, či neznělý. Jedná se o velmi důležitý blok, jelikož na jeho základě se určuje budící signál (při použití základního AR modelu). Špatná klasifikace segmentu může mít velmi špatný vliv na výslednou kvalitu syntetické řeči.

V praxi se pro detekci znělosti využívá vícero algoritmů, založené na rozdílných principech. Různé algoritmy detekce F_0 v řečovém signálu můžeme rozdělit do následujících kategorií [2]:

- Detekce v časové oblasti.
- Detekce ve frekvenční oblasti.
- Detektory založené na anatomických modelech.

3.6.1 Detektory v časové oblasti

Za pomocí těchto technik se díváme na vstupní signál jako měnící se amplitudu v čase a je snaha nalézt v křivce určitou periodicitu, nebo-li opakující se úseky.

ACF

Detektory pomocí ACF (autokorelační funkce) jsou jedny z nejvíce používaných metod odhadu základního tónu v časové oblasti [7]. Základní myšlenka autokorelační metody je, že hodnota autokorelace vypovídá o podobnosti mezi částí signálu s(k) a jeho posuvem v čase s(k+m).

Pro diskrétní signál s(k), je krátkodobá autokorelační funkce definovaná jako:

$$R_n(m) = \sum_{k=-\infty}^{\infty} s(k)w(n-k)s(k+m)w(n-k-m), \tag{3.8}$$

kde w(n) je Hammingovo nebo pravoúhlé okénko.

Při analýze pouze jednoho segmentu zpracovaného signálu, váženého pravoúhlým okénkem, můžeme vztah 3.8 upravit na:

$$R(m) = \sum_{k=0}^{N-1-m} s(k)s(k+m), \tag{3.9}$$

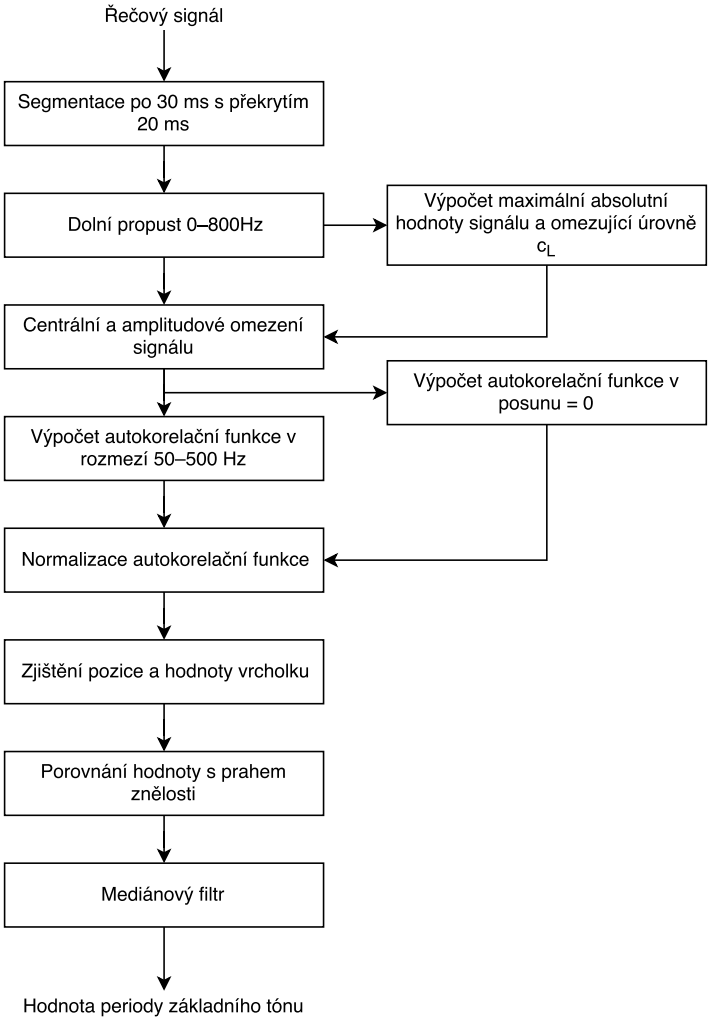
kde N představuje velikost okénka. Volí se minimálně 20–40 ms, aby segment obsahoval alespoň jednu periodu základního tónu. Hledáme takové $m > 0$, pro které je velikost R(m) maximální. Toto zpozdění symbolizuje periodu T_0.

Pokud je signál periodický, pak autokorelační funkce R(m) bude také. Maximální hodnota je v m = 0 a R(0) je rovno energii analyzovaného signálu. Harmonický signál bude mít v autokorelační funkci vrcholky v násobcích základní frekvence F_0. Tato technika je nejvíce účinná

u nízkých až středních frekvencích a je tedy obzvláště používaná v oblastech rozpoznání řeči, kde je omezený rozsah výšky hlasu [2].

MACF

Průběh autokorelační funkce zobrazuje mimo informaci o periodě základního tónu i mnoho dalších vrcholků, které jsou způsobeny formantovou strukturou. Tyto vrcholky mohou být příčinou chybného odhadu základního tónu. Modifikovaná autokorelační funkce (MACF) se liší od obyčejné autokorelační metody tím, že se signál předzpracuje s cílem potlačit formantovou strukturu pomocí centrálního, popř. centrálního a amplitudového omezovače [3].

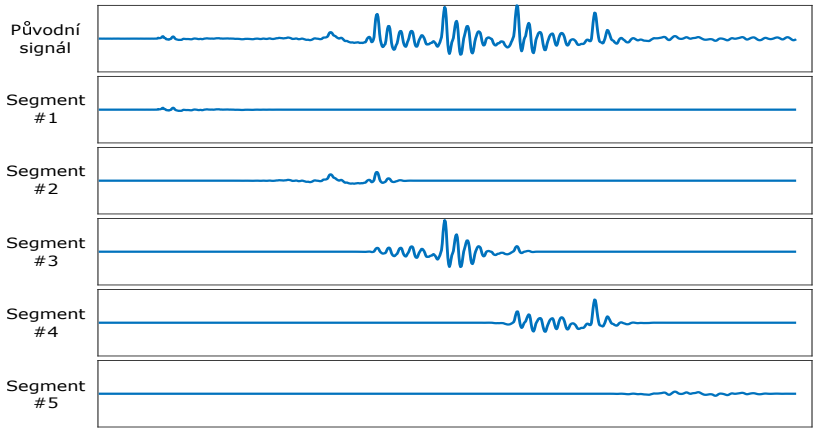


Obrázek 3.7: Blokový diagram detektoru znělosti MACF (převzané z [3]).

Jednotlivé funkční bloky algoritmu MACF z obrázku 3.7 si následně rozepíšeme pro lepší znázornění celého procesu.

Segmentace

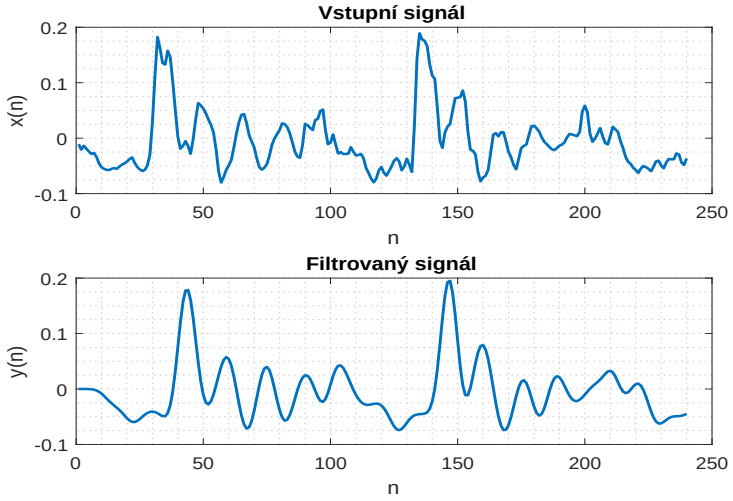
Vstupní řečový signál se rozdělí na překrývané segmenty o délce 30 ms s celkovým překrytím 20 ms. Tato segmentace je znázorněna v obrázku 3.5. Jedná se o velmi základní a nutný proces v oblasti zpracování řeči. Na následujícím obrázku 3.8 lze tento proces vidět na reálném příkladě.



Obrázek 3.8: Segmentace řečového signálu.

Dolní propust

Řečový signál se následně zbaví redundancí použitím filtru dolní propusti o mezní frekvenci $F_m = 800Hz$. Pro filtraci byl vybrán Butterworthův filtr 10. řádu, který se vyznačuje plochou amplitudovou charakteristikou a malým fázovým zkreslením. V Matlabu použijeme funkci *butter*, jejíž výstupními hodnotami jsou koeficienty polynomů tvořeného filtru. Filtrace řečového segmentu se následně provede pomocí vestavěné funkce *filter*.



Obrázek 3.9: Filtrovaný signál (DP–800 Hz).

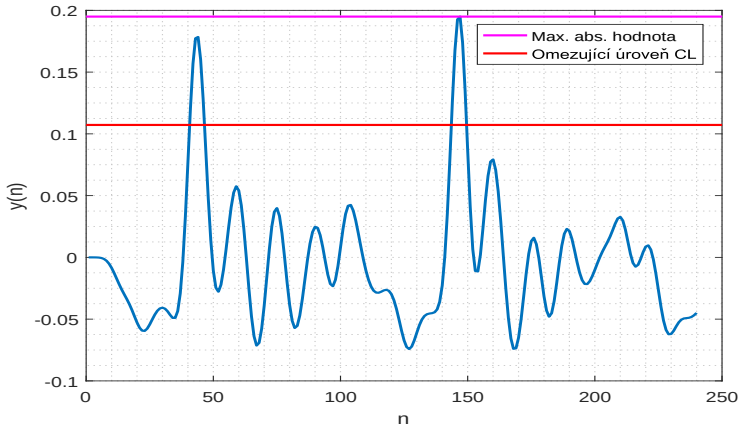
Výpočet omezující úrovně C_L

Hodnota omezující úrovně se neurčuje pouze jedním způsobem. Člověk–expert tuto úroveň může nastavit dle vlastního uvážení na hodnotu, vyhovující jeho aplikaci. Alternativně zde existuje pár používaných adaptivních algoritmů na přepočítání úrovní zvlášť pro každý řečový segment.

První algoritmus bere v potaz maximální absolutní hodnotu signálu v právě analyzovaném segmentu a omezující úroveň dostaneme prostým vynásobením této hodnoty zvolenou konstantou (např. 0.55), dle následujícího vzorce:

$$C_L = 0.55 \cdot \max |y(n)| \,, \tag{3.10}$$

kde y(n) je filtrovaný signál analyzovaného segmentu.

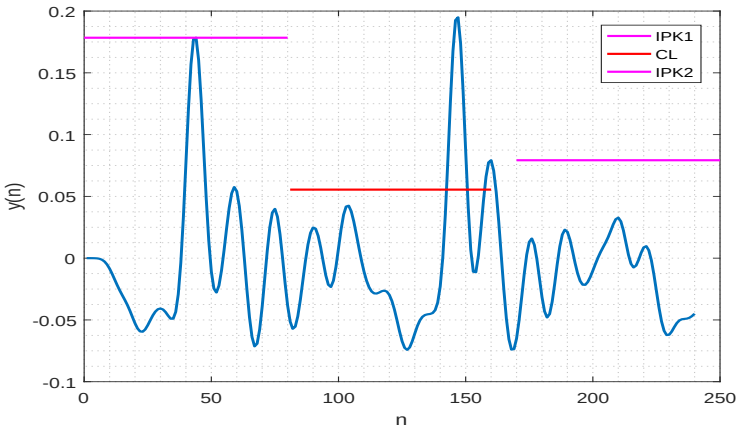


Obrázek 3.10: Výpočet omezující úrovně C_L první metodou.

Při použití druhé metody se omezující úroveň C_L spočítá dle následujícího vzorce:

$$C_L = p \cdot min \{IPK1; IPK2\} \,, \tag{3.11}$$

kde p = 0,6–0,8 a IPK1 a IPK2 jsou maximální absolutní hodnoty prvního a třetího mikrosegmentu každého rámce.



Obrázek 3.11: Výpočet omezující úrovně C_L druhou metodou.

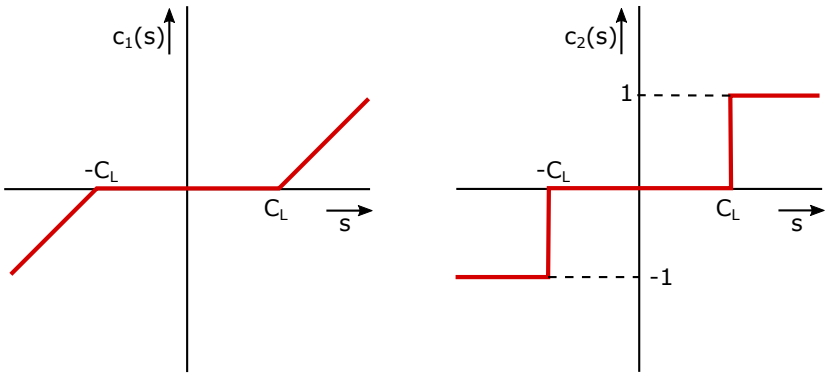
Centrální a amplitudové omezení signálu

Vstupně-výstupní funkci centrálního omezovače lze vyjádřit následovně:

$$c_1(s(k)) = \begin{cases} s(k) - C_L & \text{pro } s(k) > C_L \\ 0 & \text{pro } |s(k)| \leq C_L \\ s(k) + C_L & \text{pro } s(k) < -C_L \end{cases}, \tag{3.12}$$

Vstupně-výstupní funkci centrálního a amplitudového omezovače lze vyjádřit následovně:

$$c_2(s(k)) = \begin{cases} 1 & \text{pro } s(k) > C_L \\ 0 & \text{pro } |s(k)| \leq C_L \\ -1 & \text{pro } s(k) < -C_L \end{cases}, \tag{3.13}$$



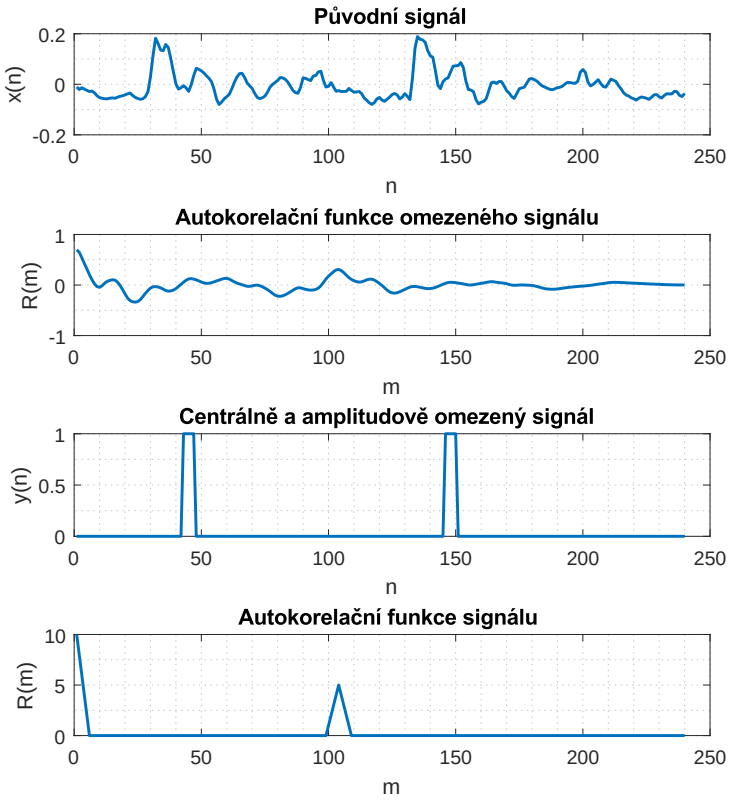
Obrázek 3.12: Vstupně-výstupní charakteristiky: centrálního omezení; centrálního a amplitudového omezení.

Výpočet autokorelační funkce

Autokorelační funkce je v tomto případě určena jako:

$$R'(m) = \sum_{k=0}^{N-1-m} c(s(k))c(s(k+m)), \tag{3.14}$$

kde N = 240 při $F_{vz} = 8$ kHz. Tato funkce je následně normalizována a určeno její maximum pro m = 16–160. Tyto hodnoty byly vypočítány, aby odpovídaly frekvenci základního tónu v rozmezí 50–500 Hz, při vzorkovací frekvenci 8 kHz (např. $\frac{F_{vz}}{50} = \frac{8000}{50} = 160$). Příslušný mikrosegment je stanoven jako znělý, pokud toto maximum překročí určitou prahovou hodnotu (např. 0,4). Podle pozice m se stanoví perioda základního tónu [11].



Obrázek 3.13: Porovnání ACF a MACF detektorů F_0 u znělého segmentu.

3.6.2 Detektory ve frekvenční oblasti

CPD

Detekce znělosti ve frekvenční oblasti z kepstrální analýzy CPD (Cepstrum Pitch Determination) má své výhody oproti autokorelačním technikám. Kepstrum získáme jako výsledek inverzní Fourierovy transformace logaritmu spektra signálu. U znělých řečových segmentů se vyznačuje silným vrcholkem, který odpovídá periodě základního tónu [9]. Předpokládáme, že signál znělé řeči s(n) může být zapsán jako:

$$s(n) = e(n) * h(n), \tag{3.15}$$

kde e(n) je excitační signál a h(n) je impulsní odezva hlasového traktu. V autokorelační funkci je mezi zdrojovým signálem a hlasovým traktem provedena konvoluce. Výsledkem jsou v autokorelační funkci široké vrcholky a v některých případech i vícero vrcholků [3]. Ve frekvenční oblasti se z konvoluce stává násobení a ze vztahu 3.15 se stává:

$$S(\omega) = E(\omega) \cdot (H(\omega)), \tag{3.16}$$

kde $S(\omega) = F\{s(n)\}$, $E(\omega) = F\{e(n)\}$ a $H(\omega) = F\{h(n)\}$.
Rovnice 3.16 může být přepsaná jako:

$$F^{-1}\{log[S(\omega)]\} = F^{-1}\{log[E(\omega)]\} + F^{-1}\{log[H(\omega)]\}. \tag{3.17}$$

Vztah mezi zdrojovým signálem a hlasovým traktem se změnil na součet. Je možné oddělit část kepstra, která reprezentuje zdrojový signál a najít pravou periodu základního tónu. Z tohoto důvodu je odhad základního tónu z kepstra přesnější, než přes autokorelační techniky [7]. Pro zjištění základního tónu postačí pouze reálná část kepstra, které je pro diskrétní signál s(n) definováno jako:

$$C(m) = \frac{1}{N} \left\| \sum_{k=0}^{N-1} S(k) \cdot e^{-j \cdot \frac{2\pi}{N} mk} \right\|, \tag{3.18}$$

kde S(k) je logaritmické amplitudové spektrum s(n):

$$S(k) = \log \left\| \sum_{k=0}^{N-1} S(n) \cdot e^{-j \cdot \frac{2\pi}{N} mk} \right\|. \tag{3.19}$$

Kepstrum se skládá z vrcholku, který se vyskytuje ve vysoké kvefrenci, která je rovna periodě základního tónu v sekundách a z nízkokvefrenčních informací, odpovídajících formantní struktuře v logaritmickém spektru [9]. Abychom získali odhad frekvence základního tónu F_0, hledáme vrcholek v rozmezí kvefrencí, odpovídající typickým frekvencím F_0. Tento detektor je možný kombinovat s detektorem ZCR pro upřesnění klasifikace segmentů [8].

4. Syntéza řeči

4.1 Druhy syntézy

Mezi základní druhy syntézy řeči patří:

- **Artikulační metoda.** Patří mezi nejsložitější metody při tvorbě syntézy řeči, protože se jedná o kompletní modelování hlasového traktu. Výpočetně je velmi náročná a momentálně nedosahuje významných úspěchů. Slouží pouze pro pedagogické účely.
- **Formantová metoda.** Je založená na akustické teorii vytváření řeči. Jedná se o zjednodušenou simulaci tvoření řeči člověkem. Řečový signál je tvořený pomocí stanovených pravidel. Základním prvkem jsou rezonátory, které slouží pro modelování spektrálních vrcholků. Obsahuje zdroj buzení, který se skládá z generátoru impulzů pro znělé zvuky a pro neznělé zvuky bílý šum nebo smíšené buzení. Dále obsahuje hlasový trakt, který je modelovaný pomocí filtru, jehož parametry reprezentují odezvu hlasového traktu. Jednalo se o velmi úspěšnou a používanou metodu. Vyznačovala se jednoduchostí modelu, jednoduché řazení prozodických charakteristik, stálá kvalita a schopnost vytvářet kvalitní řeč. Jako jeho nevýhody bylo například vytváření některých zvuků a nízká přirozenost řeči (velmi robotické).
- **Konkatenační metoda.** Tato metoda předpokládá, že se řeč skládá z jednotlivých řečových jednotek. Celý řečový signál se segmentuje na vybrané jednotky, které jsou následně uloženy do inventáře pro pozdější rekonstrukci. Výsledná syntetická řeč se získá zřetězením (konkatenací) jednotlivých řečových jednotek.

4.2 Konkatenační syntéza

Vytváření inventáře řečových jednotek lze vytvářet buď ručně, dle uvážení člověka–experta nebo pomocí automatických algoritmů, například pomocí Skrytých Markovův modelů (HMM). Řečové jednotky lze reprezentovat buď neparametricky, tedy přímo uloženými vzorky řeči nebo parametricky, kde se výstupní signál rekonstruuje průchodem budícího signálu tvořeným filtrem, popsaným uloženými LPC koeficienty. Pro minimalizaci nespojitostí na hranicích řetězených jednotek lze také uplatňovat spektrální modifikace jednotek.

4.2.1 Výhody a nevýhody konkatenanční syntézy

Jde zdaleka o nejpoužívanější přístup syntézy řeči. Samozřejmě má i přes mnoho výhod určité slabé stránky. Neustále se ale pracuje na jejich vylepšení.

Mezi výhody této techniky patří:

- **Kopírování hlasu řečníka.** Ze svého principu tato technika kopíruje hlas osoby, která namluvila řečový korpus (slova, věty, logatomy,...). Korpus se později analyzuje a je z něj vytvořený inventář řečových jednotek. Analyzujeme skutečné, přirozené segmenty řeči a přispívá to tedy k výsledné přirozenosti tvořeného signálu. Není nutné vytvářet syntetickou řeč pomocí složitých pravidel.
- **Odpadá nutnost znalosti procesu vytváření řeči.** Jelikož pracujeme s reálnými segmenty řeči, není nutné znát detailní proces, jak vlastně řeč vzniká.
- **Rychlý návrh systému.** Parametry filtrů jsou získány automaticky z analyzovaného řečového segmentu, není tedy nutné hledat složitá pravidla pro syntézu, jako např. u formantové syntézy. Časová náročnost této techniky spočívá především v namluvení řečového korpusu a jeho označkování (vytvoření inventáře jednotek).
- **Vysoká kvalita.** Tato technika se vyznačuje vysokou kvalitou syntetické řeči. Záleží vysoce na použitém modelu a také na počtu reprezentovaných řečových jednotek. Složitější systémy mívají vícero zastoupení každé jednotky a dosazují se na základě jejich kontextu. S dobře realizovaným algoritmem výběru jednotek mohou mít velmi přirozenou výstupní řeč a vykazují lepší kvalitu řeči, než syntetizéry s jedním zastoupením jednotky. Ty mají sice horší kvalitu řeči, ale jsou pořád hodnoceny lépe, než systémy založené na formantové syntéze [11].

Mezi nevýhody této techniky patří:

- **Nebezpečí nestálé kvality.** Může se stát, že daná jednotka nepasuje do daného kontextu i přes vhodně vytvořený algoritmus výběru jednotek a výsledné slovo, či jeho část bude znít špatně. Je samozřejmě možné, že v našem omezeném řečovém korpusu může určitý kontext jednotky chybět a výběr nejlepší jednotky je stejně pro danou situaci špatný. Nebezpečí vzniku takové chyby můžeme redukovat vhodným výběrem namluvených vět takovým způsobem, aby byly jednotky reprezentovány ve všech spektrálních a prozodických kontextech.
- **Oblast řetězení.** Princip konkatenační syntézy – nutnost řetězit jednotky zůstává jejím největším nedostatkem a oblast řetězení může představovat riziko problémů. Pro redukci těchto problémů existují metody spektrálního vyhlazení přechodů jednotek lineární interpolací parametrů filtrů řetězených difonů a dále již zmíněné systémy, využívající výběr nejvhodnějšího zastoupení jednotky z inventáře. Systém, který podporuje

výběr i delších řečových jednotek (fonémy, slova, či celé věty) zmenšuje počet oblastí řetězení a zlepšuje kvalitu řeči.

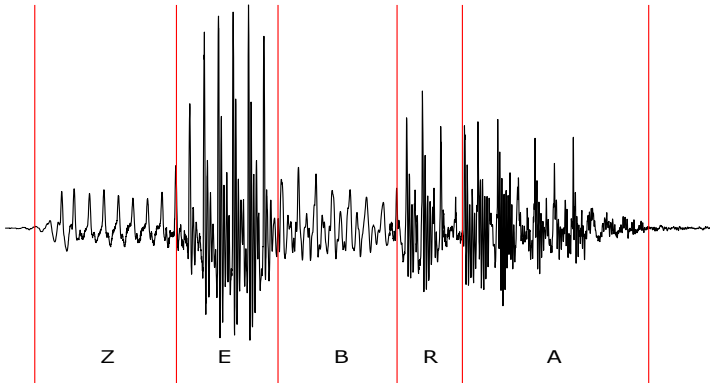
- **Náročnost na paměť.** V případě korpusově orientované konkatenační syntézy, která obsahuje vícero reprezentací jednotek, může mít značné nároky na paměť. Velmi rozsáhlé databáze mohou dosáhnout stovek, až tisíce MB.

- **Změna hlasu.** Princip této metody kopíruje hlas osoby, která namluvila řečový korpus. V předchozí části byla tato vlastnost zařazena jako výhoda systému. Dá se ale brát i jako negativní stránka a to v tom, že neumožňuje jednoduchou změnu hlasu na jiného řečníka. Řešení může být namluvení tolika databází, kolik chceme rozdílných hlasů, je ale nutné myslet na zvýšené požadavky na paměť a dobu vývoje systému (nahrání dalších korpusů, jejich analýza, možnost výběru hlasů,...). V opačném případě lze použít určitých parametrických metod úpravy řeči, abychom dosáhli požadovaného hlasu. Nejedná se však o triviální záležitost a je riziko zkreslení a degradace výsledné řeči.

4.2.2 Volba řečových jednotek

Fonémy

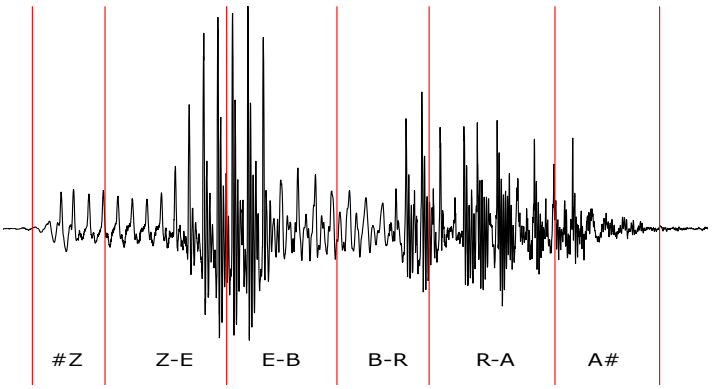
Foném je základní stavební jednotka řeči. Výhodou jejich použití pro syntézu řeči je nízký počet jednotek, který je závislý na konkrétním jazyku (pro český jazyk jich je 42), tím pádem je velmi snížená práce vytváření inventáře řečových jednotek a také se o hodně snižuje velikost kterou zabírají v databázi. Nevýhodou ale je, že neobsahují koartikulační bod, kde se jednotlivé fonémy prolínají a tedy výsledná řeč nepůsobí přirozeně [8].



Obrázek 4.1: Vyznačení hranic fonémů ve slově *ZEBRA*.

Difóny

Difóny jsou nejpoužívanější stavební jednotky při tvorbě umělé řeči. Jejich největší výhodou je to, že obsahují přechod mezi jednotlivými fonémy (koartikulační bod). Každý difón začíná v polovině předchozího fonému a končí v polovině následujícího fonému. Pro syntézu řeči je ale zapotřebí mnohem více jednotek v inventáři než pouze fonémů. Jestliže se v konkrétním jazyku nachází N počet fonémů, pak je v daném jazyce N^2 difónů. Pro český jazyk by to tedy teoreticky vycházelo na $49^2 = 2401$ difónů. V praktickém využití se jich ale používá méně, jelikož celkové číslo zahrnuje i kombinace hlásek, které se v češtině nevyužívají, například 'xh' nebo redundantní hlásky, jako například 'y', které se dá nahradit hláskou 'i' nebo hláska 'x', která je kombinací hlásek 'k' a 's'. Reálně bychom se tedy přiblížili počtu okolo 1200 difónů.



Obrázek 4.2: Vyznačení hranic difónů ve slově *ZEBRA*.

Trifóny

Trifóny se skládají ze tří fonémů. Začínají v polovině prvního fonému a končí v polovině třetího fonému. Jeho výhodou oproti difónům je lepší zachování koartikulačních jevů mezi hláskami. Nevýhodou je ale zase potřeba vyššího počtu analyzovaných jednotek na reprezentaci jazyka. Pro český jazyk by jejich počet teoreticky vycházel na $49^3 = 117649$ trifónů. Jejich plný počet se jich nevyužívá, jedná se stále ale o velmi rozsáhlý inventář na který je už zřejmě jistě zapotřebí použití automatické segmentace.

4.2.3 Inventář řečových jednotek

Pro analýzu je potřeba vybrat taková slova, které zahrnují všechny kombinace hlásek a reprezentovaly by tedy veškeré řečové jednotky.

Řečový korpus

Pro realizaci TTS systému je nejdříve velmi důležité si uvědomit oblast použití. Na základě oblasti použití, může systém syntézi řeči pracovat buď s omezeným slovníkem, kde jsou v inventáři uložená celá slova, celé věty. Taková syntéza je velice účinná s velmi přirozeným projevem, hodí se ale pouze pro velmi specifické oblasti použití - např. oznámení na vlakovém nádraží, kde není nutná jistá flexibilita TTS systému, jelikož všechna oznámení i názvy stanic jsou již předdefinovány a pouze se kombinují.

Pro tvoření libovolných slovních spojení jsou vhodnější menší jednotky, například fonémy, difóny nebo trifóny na usnadnění již velmi časově náročného procesu vytváření inventáře řečových jednotek a odlehčení nároků na paměť.

Segmentace

Pojem segmentace představuje hledání hranic akustických realizací řečových jednotek v promluvách. Obecně se dá rozdělit na dva druhy:

- Ruční segmentace.
- Automatická segmentace.

Ruční segmentace je proces, kde člověk–expert označuje hranice řečových jednotek ručně a dle vlastního uvážení pouze na základě zobrazeného signálu v čase či spektrogramu. Segmentace se tradičně provádí ručně, ale jedná se o velmi časově náročný a vyčerpávající proces. Samozřejmě i velmi trénované osoby mohou mít jiné představy, kam určit hranici a také se nemohou vyvarovat určitých chyb označení při zpracování velkého množství dat. Díky velkému rozvoji konkatenační syntézy, použitím stále větších a rozsáhlejších inventářů a využitím menších řečových jednotek (např. trifonů), je zapotřebí rozvíjet algoritmy pro automatickou segmentaci.

4.2.4 SAMPA

Přepis promluvy a symboly, pod kterými se dané řečové jednotky budou ukládat musí být dané nějakou standardní fonetickou abecedou. Je na výběr vícero soustav značek, např. mezinárodní fonetická abeceda *IPA*, která poskytuje úplnou soustavu fonetických značek, není ale příliš vhodná pro reprezentaci symbolů v počítači. Vznikla tedy nová fonetická abeceda zvaná **SAMPA** (Speech Assesment Methods Phonetic Alphabet), která je v počítači snadno zobrazitelná. V této práci byla použita lehce zjednodušená *SAMPA* abeceda pro reprezentaci jednotek, dle následující tabulky 4.1.

Tabulka 4.1: SAMPA tabulka (převzato z [11]).

	SAMPA	Zjednodušená SAMPA	Slovo	Transkripce		SAMPA	Zjednodušená SAMPA	Slovo	Transkripce
vokály	i	i	lis	lis	plozivy	p	p	pec	pets
	e	e	pes	pes		b	b	bratr	bratr
	a	a	sad	sad		t	t	tuk	tuk
	o	o	kov	kov		d	d	dům	du:m
	u	u	sukně	sukNe		c	T	tělo	Telo
	i:	i:	víno	vi:no		J\	Dj	děda	djeda
	e:	e:	lék	le:k		k	k	kost	kost
	a:	a:	sál	sa:l		g	g	tygr	tigr
	o:	o:	kód	ko:d	nazály	m	m	muž	muZ
	u:	u:	růže	ru:Ze		n	n	nos	nos
diftongy	o_u	ou	bouda	bouda		J	N	laňka	laNka
	a_u	au	auto	auto	afrikáty	t_s	ts	cena	tsena
	e_u	eu	euro	euro		t_S	tS	oči	otSi
frikativy	f	f	fík	fi:k		d_z	ts	leckdo	letskdo
	v	v	vítr	vi:tr		d_Z	dZ	džbán	dZba:n
	s	s	sůl	su:l	významné alofony	N	n	tango	tango
	z	z	koza	koza		F	m	tramvaj	tramvaj
	S	S	škola	Škola		G	x	abych byl	abix bil
	Z	Z	žena	Zena		Q\	R	tři	tRi
	x	x	chata	xata		r=	r	krk	krk
	h\	h	hůl	hu:l		l=	l	vlk	vlk
	l	l	vlak	vlak		m=	m	osm	osm
	r	r	rok	rok		?		Ano.	ano
	P\	R	moře	moRe		@		www	v.v.v.
	j	j	jev	jev					

5. Kódování řeči

Z analyzovaných segmentů jsou informace o použitém filtru a budícím signálu zakódovány pomocí určitých parametrů. Jeden z nejpoužívanějších modelů vokálního traktu je založený na technice tzv. lineární predikce. Budící signál těchto filtrů u základního modelu je velmi zjednodušený a považuje části řeči pouze jako čistě znělé, či neznělé. Již z principu nedosahuje přirozenost řeči vysoké úrovně a vyznačuje se velmi bzučivým, až robotickým zvukem. Tento model lze vylepšit metodami RELP a RPE-LPC, která budící signál nevytváří dle parametrů, ale je přenášen spolu s koeficienty filtru.

5.1 Lineární predikce

Lineární predikce (LP) je jednou z nejvíce využívaných technik v kódování řeči, syntéze řeči, rozpoznání řeči a verifikaci řečníka a na uskladnění řečových signálů [10].

LP nabízí velmi přesné a efektivní odhady řečových parametrů [6]. Základní myšlenka lineární predikce je, že vzorek řeči může být aproximován jako lineární kombinace předešlých vzorků, např.

$$s(n) = \sum_{k=1}^{p} \alpha_k s(n-k), \tag{5.1}$$

$$s(n) = \sum_{k=1}^{p} a_k s(n-k) + Gu(n), \tag{5.2}$$

kde k je časový index, p znázorňuje počet koeficientů v modelu (řád prediktoru), a_k, k = 1,...p jsou definované jako koeficienty lineární predikce, G je intenzita (gain) systému a u(n) je excitační signál, který může být buď kvaziperiodický sled impulsů nebo bílý šum.

Originální účel LPC bylo modelovat produkci lidské řeči. Zdrojový signál modeluje hlasivky, zatímco rezonantní filtr *h(n)* modeluje vokální trakt. Výsledný signál je,

$$s(n) = h(n) * e(n). \tag{5.3}$$

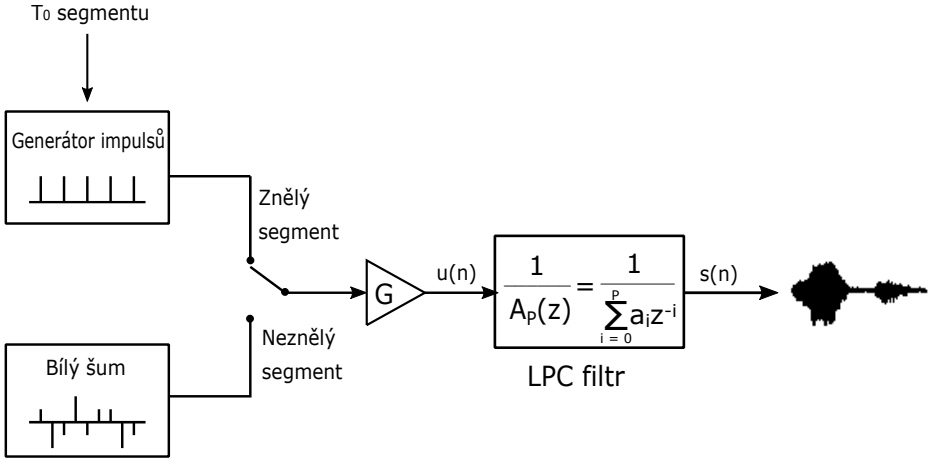
Parametry a_k rozhodují o spektrálních charakteristikách daného zvuku pro každý z těchto dvou typů excitačního signálu a jsou široce používány v mnoha systémech kódování řeči a systémech automatického rozpoznání řeči.

Rovnice 5.3 může být přepsána ve frekvenční oblasti pomocí z-transformace. Pokud H(z) je přenosová funkce systému pak dostaneme:

$$H\left(z\right) = \frac{G}{1 + \sum_{k=1}^{P} a_k z^{-1}} = \frac{G}{A_{(z)}}. \tag{5.4}$$

5.1.1 Jednoduchý AR model

Na obrázku 5.1 lze vidět jednoduchý autoregresní (AR) model lineární predikce. Jako excitační (budící) signál v tomto modelu může sloužit buď sled impulzů nebo bílý šum. Rozhoduje se na základě, zda je analyzovaný rámec řečového signálu znělý, či neznělý.



Obrázek 5.1: Autoregresní model LP (převzaný z [11]).

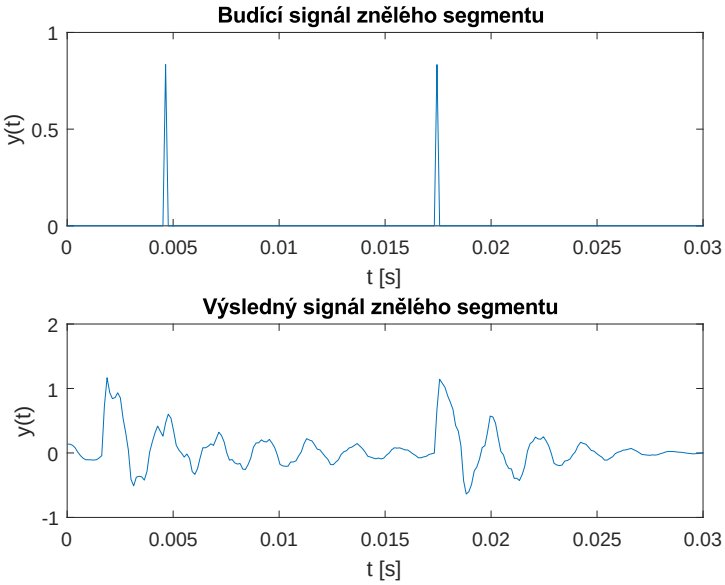
Klasický AR model syntézy řeči pomocí koeficientů lineární predikce je řízený následujícími parametry:

- **Znělost**. Rozhodnutí, zda analyzovaný řečový segment je znělý (např. samohlásky a, e, i, o, u a některé souhlásky, např. j, k, r), či neznělý (většina souhlásek, například 'm').
- **Intenzita segmentu** (gain). Energie analyzovaného segmentu.
- **Koeficienty filtru**. Na jejich základě se tvoří filtr, reprezentující vokální trakt.
- **Perioda** T_0. Časový úsek mezi po sobě následujícími budícími pulsy (pouze pro znělé segmenty).

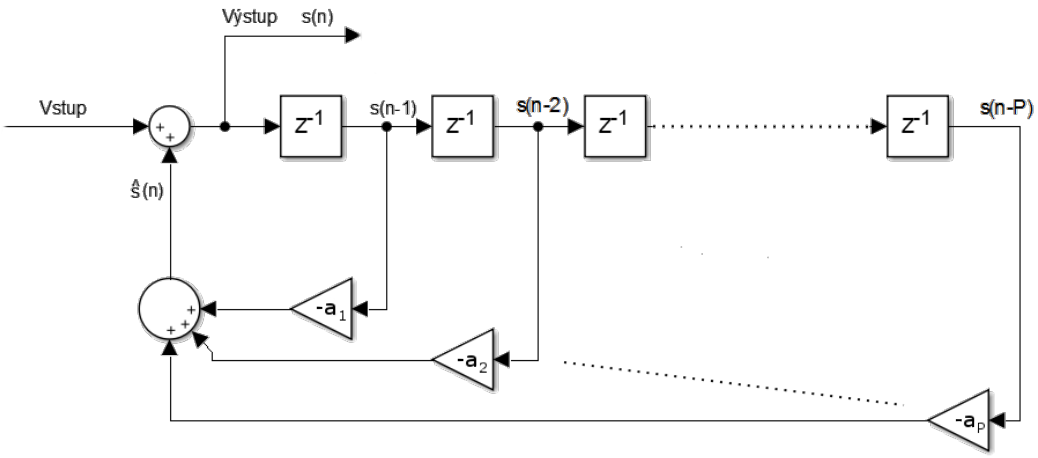
Tento způsob kódování umožňuje efektivní reprezentaci řečového signálu. S jeho pomocí se nemusí uchovávat celé řečové segmenty, pouze jeho parametrický popis a signál se rekonstruuje na cílové stanici.

Jako budící signál pro neznělý segment se použije bílý šum s rovnoměrným rozdělením. Generovaný šum má jednotkovou varianci s upravenou velikostí intenzity (gain). Každá instance generování takového segmentu je vždy v časové oblasti jiná, zní však podobně.

Filtr pro znělé segmenty je buzený signálem, který se skládá ze sledů impulsů s jednotkovou amplitudou a škálovaný intenzitou (gain) aby úroveň energie byla stejná jako v původním řečovém signálu.



Obrázek 5.2: Tvořený umělý znělý segment.



Obrázek 5.3: Blokový diagram syntetického filtru LPC.

5.1.2 RELP

Metoda RELP (z anglického Residual Excited Linear Prediction) vytváří umělou řeč na základě buzení filtru kvantizovaným zbytkovým signálem. Jedná se o velmi využívanou metodu v syntéze a kódování řeči a vyznačuje se velmi dobrou kvalitou.

Reziduální signál je v podstatě chyba lineární predikce. Hlasové ústrojí je modelováno systémem s přenosovou funkcí:

$$H\left(z\right) = \frac{G}{1 + \sum_{k=1}^{P} a_k z^{-1}}. \tag{5.5}$$

Znělé segmenty řeči jsou buzeny posloupností pulsů o frekvenci F_0 a bílým šumem pro neznělé segmenty. Výstupní signál filtru je dán:

$$s(n) = -\sum_{k=1}^{p} a_k s(n-k) + Gu(n). \tag{5.6}$$

Za předpokladu, že řeč je zpracována lineárním prediktorem, platí rovnice:

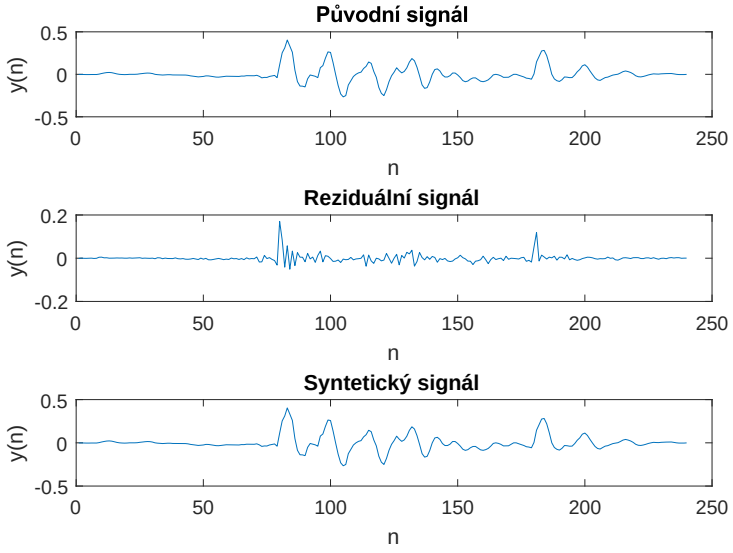
$$\hat{s}(n) = \sum_{k=1}^{p} \alpha_k s(n-k) \tag{5.7}$$

a chyba predikce je nyní definována jako:

$$e(n) = s(n) - \hat{s}(n) = s(n) + \sum_{k=1}^{p} \alpha_k s(n-k). \tag{5.8}$$

Když srovnáme rovnice 5.6 a 5.8, zjistíme, že pokud je $\alpha_k = a_k$, pak $e(n) = Gu(k)$. Reziduální signál tedy obsahuje informace o budící funkci systému a je z něj možné získat důležité informace, jako např. hodnotu základní frekvence F_0 nebo jej lze použít k určení znělosti analyzovaného segmentu. Z rovnice 5.8 lze také pozorovat, že reziduální signál je v podstatě výstupem filtru buzený vstupním signálem s(n) a s přenosovou funkcí:

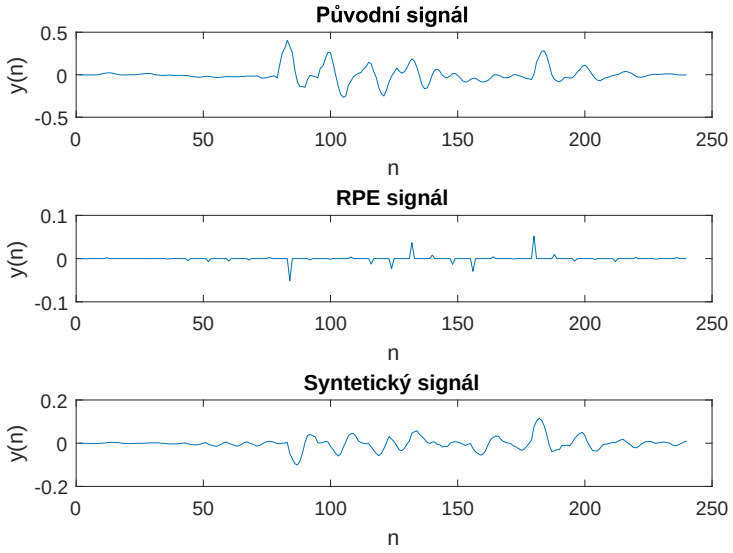
$$A(z) = 1 + \sum_{k=1}^{P} a_k z^{-1}. \tag{5.9}$$



Obrázek 5.4: Syntéza pomocí techniky RELP.

5.1.3 RPE-LPC

Technika RPE-LPC (Regular-pulse excited LPC) reprezentuje reziduální signál určitým počtem impulsů na daný rámec dat. Často se používá redukce v poměru 8:1. To znamená, že pro rámec dat o velikosti např. 240 vzorků bude reziduální signál modelován pouze 30 stejně vzdálenými pulsy. Podobně jako u techniky RELP se nemusí přenášet koeficienty *pitch* a hodnoty intenzity rámce G.



Obrázek 5.5: Syntéza pomocí techniky RPE-LPC.

5.1.4 MPE-LPC

Tato technika pracuje na podobném principu jako RPE-LPC. Rozdíl v nich je pouze v tom, že zvolené impulsy, které mají reprezentovat plný reziduální signál nejsou zvoleny ve stejné vzdálenosti od sebe. Jejich pozice a amplitudy jsou vybrány, aby byly nejvěrnější reprezentací chyby predikce.

5.1.5 Levinson-Durbinův algoritmus

Výpočet koeficientů filtru a_i, k_i a zesílení G lze provést pomocí vysoce účinného iterativního algoritmu navržen Levinsonem a modifikovaném Durbinem [11]. Řešení je vyjádřeno pro i=1, 2, ..., Q jako:

$$\begin{aligned} &E_n^{(0)} = R_n\left(0\right), \\ &k_i = -[R_n(i) + \sum_{j=1}^{i-1} a_j^{(i-1)} R_n(i-j)]/E_n^{(i-1)}, \\ &a_i(i) = k_i, \\ &a_j^{(i)} = a_j(i-1) + k_i a_i - j^{(i-1)}, \qquad 1 \leq j \leq i-1, \\ &E_n^{(i)} = (1-k_i^2) E_n^{(i-1)}, \end{aligned} \tag{5.10}$$

kde $a_j^{(i)}$ je j-tý parametr prediktoru řádu i, k_i jsou tzv. koeficienty odrazu (v anglické literatuře také známé jako PARCOR), $E_n^{(i)}$ vyjadřuje hodnoty chyby predikce.

Za předpokladu, že budící signál systému je buď sled impulsů nebo bílý šum, lze intenzitu G vypočítat jako:

$$G^2 = R_n(0) + \sum_{i=1}^{Q} a_i R_n(i) = E_n, \tag{5.11}$$

5.1.6 Koeficienty filtru

Koeficienty lineární predikce mohou být reprezentovány v různých podobách. Jako základní bývají považovány přímo koeficienty filtru a_i.

Koeficienty a_i

Vycházejí přímo z algoritmu 5.10. Obecnou syntézu řeči podle teorie zdroj a filtr můžeme napsat jako:

$$S(z) = H(z)E(z), \tag{5.12}$$

kde E(z) je budící (excitační) signál a H(z) je celopólový syntetický LPC filtr. Syntetický signál s(n) můžeme poté spočítat jako:

$$s(n) = p(n) - \sum_{i=1}^{P} a_i s(n-i), \tag{5.13}$$

kde:

$$p(n) = Gu(n). \tag{5.14}$$

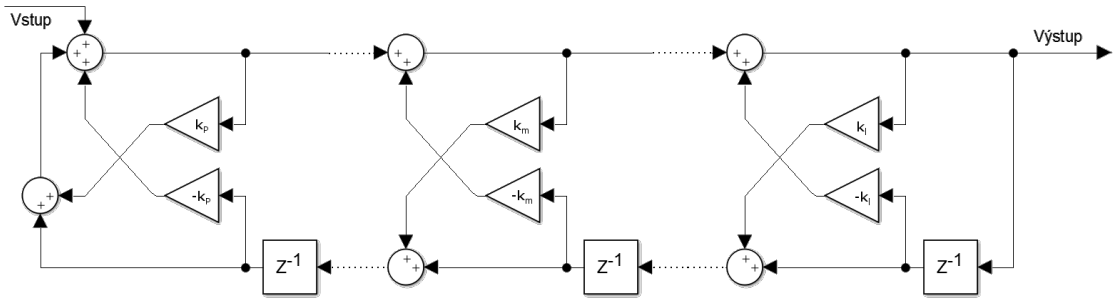
Nabízí se přímé použití filtru $1/A_p(z)$, nicméně je problém se stabilitou takto tvořených filtrů. Přímý přenos jeho koeficientů je nechtěný kvůli chybám kvantizace a není ani vymezeno žádné kritérium na rozsah těchto koeficientů pro zaručení stability filtru [5].

PARCOR

Koeficienty k_i pro $1 \leq i \leq P$ slouží jako alternativní reprezentace koeficientů lineární predikce. V literatuře mohou být známy pod více názvy, např. reflexní koeficienty, koeficienty odrazu. Jsou mnohem méně náchylné na kvantizační chyby než koeficienty a_i a jsou tedy více využívané v praktických úlohách jako třeba kódování nebo komprese řeči.

Pokud velikost reflexních koeficientů $|k_i|$ je méně než 1 pro $1 \leq i \leq P$, pak všechny kořeny polynomu $A(z) = 1 - \sum_{k=1}^{P} \alpha_k^P z^{-k}$ leží uvnitř jednotkového kruhu. To znamená, že pokud $|k_i| < 1$, pak u tvořeného filtru H(z) bude zaručena stabilita.

V syntéze řeči pomocí PARCOR koeficientů se používá speciálně navržený filtr s křížovou strukturou.



Obrázek 5.6: Blokový diagram syntetického filtru LPC s křížovou strukturou.

Je možné přepočítat PARCOR koeficienty na koeficienty a_i dle následujících vztahů [11]:

$$\begin{aligned} a_i^i &= k_i, & i &= 1, ..., P, \\ a_j^i &= a_j^{i-1} + k_i a_{i-1}^{i-1}, & 1 &\leq j < i. \end{aligned} \tag{5.15}$$

LAR

Tyto koeficienty se využívají především pro kódování řeči, jelikož rozdělení jejich parametrů g_i je více rovnoměrné než PARCOR [11].

$$g_i = ln \frac{1 - k_i}{1 + k_i} \tag{5.16}$$

Z LAR koeficientů lze možné získat inverzním výpočtem k rovnici 5.16, kde dostaneme zpětně koeficienty PARCOR:

$$k_i = \frac{1 - exp(g_i)}{1 + exp(g_i)} \tag{5.17}$$

5.2 Prozodické a spektrální modifikace

V lidské řeči se vyskytuje velmi velká variabilita prozodických a fonetických charakteristik. Velký řečový korpus tedy nemusí zaručovat, že v něm budou zahrnuty všechny řečové kontexty. Spektrální a prozodické modifikace umožňují větší flexibilitu při použití menší databáze řečových jednotek, pro případy, kde uložené jednotky neodpovídají požadovanému syntetickému kontextu. Vlastnosti řetězených jednotek je nutno nějakým způsobem modifikovat, aby lépe popisovaly požadovaný kontext.

5.2.1 Prozodické modifikace

Umožňují změnu prozodických vlastností řetězených řečových jednotek (výšku hlasu, dobu trvání) pro nejbližší přiblížení požadovanému prozodickému kontextu řeči bez ovlivnění in-

formace a přirozenosti řeči. Jejich provedení je velmi závislé na použité metodě syntézy řeči.

V obecném AR modelu se prozodie upravuje jednoduchým způsobem. Výška hlasu je přímo uložený parametr modelu ve formě periody základního tónu T_0. Lze ho tedy upravovat přímo aplikováním faktoru, který danou periodu zvětší, či zmenší. Délku trvání segmentu je možné upravit také jednoduchým způsobem. Je předem daná velikost synteticky vytvářeného segmentu (N = 240 vzorků při F_{vz} = 8000 Hz). Na každý tento segment aplikujeme faktor, který jednotlivé segmenty rovnoměrně prodlouží, či zkrátí. Intenzita hlasu G se modifikuje dle vzorce:

$$\tilde{G} = G\sqrt{\frac{T_0}{\tilde{T_0}}}, \tag{5.18}$$

kde G je zesílení hlasu původního segmentu, $\tilde{G}$ je zesílení hlasu nového segmentu, T_0 je perioda základního tónu původního segmentu, $\tilde{T_0}$ je perioda základního tónu nového segmentu.

Prozodické úpravy řečového signálu založeného na ostatních LPC modelech jsou nejjednodušeji řešené použitím techniky jako například TD-PSOLA v časové oblasti a provedení zpětného skládání signálu ve výsledný prozodicky upravený signál.

5.2.2 Spektrální modifikace

Slouží pro minimalizaci spektrálních nespojitostí v místě řetězení řečových jednotek mezi sousedními segmenty upravením frekvenčních vlastností a vyhlazením jejich přechodu. Každé z řetězených jednotek tvořené syntetické řeči, jsou popsané posloupností vektorů zvolených parametrů p = $p_1, p_2, p_3, ..., p_N$, kde N je řád filtru. Tyto parametry mohou být např. koeficienty PARCOR. Levý a pravý řetězený segment si označíme dolním indexem L a P. Parametry na konci levého segmentu a začátku pravého segmentu označme p_L^0 a p_P^0.

V místě řetězení můžeme díky akustické podobnosti řetězených segmentů použít vyhlazování pomocí lineární interpolace, dle následujících vzorců [4]:

$$\begin{aligned} p_L^{-i} &= p_L^{-i} + (p_P^0 - p_L^0)\frac{M_L - i}{2M_L}, \qquad i = 0, 1, ..., M_L - 1 \\ p_P^j &= p_P^j + (p_P^0 - p_L^0)\frac{M_P - j}{2M_P}, \qquad j = 0, 1, ..., M_L - 1, \end{aligned} \tag{5.19}$$

Interpolace rozloží rozdíl $(p_P^0 - p_L^0)$ mezi M_L vektorů parametrů a M_P vektorů parametrů. Počet použitých vektorů obou segmentů pro interpolaci bývá obvykle různý a to v závislosti na typu použitých jednotek (hlásky, difóny,...). Obvykle se bere 4 nebo 5 vektorů nejblíže místa řetězení [11].

6. Příklady dostupných TTS systémů

SpeechTech TTS

Syntetické hlasy od společnosti SpeechTech je možno používat pro domácí nebo komerční použití. Webová stránka výrobce tvrdí, že jejich program využívá Hasičský záchranný sbor České Republiky pro vyhlašování poplachů na většině stanic. Balíčky zahrnutí celkově 7 hlasových možností, 6 v češtině a 1 ve slovenském jazyce. Verze pro vyzkoušení produktu má dostupný 1 hlas a je platná na 1 měsíc. Program disponuje podporou rozhraní Microsoft SAPI 5, syntézi řeči v *cloudu* na jejich serverech a mají také mobilní alternativu na stažení z obchodu Google Play pro telefony s OS Android [13].

KobaSpeech

KobaSpeech je hlasový syntetizér od společnosti KOBA Vision, který disponuje českými hlasy Zuzana a Iveta. Hlasy lze propojit s jakýmkoliv systémem který používá standardní rozhraní pro řeč SAPI 5 v OS Windows. Hlasy jsou vzorkované frekvencí 22 kHz a každý hlas zabírá kolem 100 MB. Software lze používat 30 dní ve zkušební verzi, po které je nutno používané hlasy zakoupit [14].

Acapela Infovox 4

Infovox je hlasový syntetizér, který disponuje českým hlasem Eliška. Jako ostatní podporuje rozhraní SAPI 5. Vypadá to, že společnost nemá k dispozici zkušební verzi na omezený čas, pouze ukázku vybraného hlasu TTS na jejich webové stránce. Program neprodávají přímo, ale skrze mezinárodní síť distributorů. V České Republice je software prodáván firmami GALOP s.r.o. a Spektra v.d.n. [15].

EPOS

Epos je open-source systém pro syntézu řeči řízené pravidly a nezávislé na jazyku, vyvinutý na Karlově Univerzitě. Toto programové řešení bylo původně vyvinuto jako nástroj pro výzkum a nabízí velmi flexibilní nastavení všech parametrů. Bylo pro něj vytvořeno několik mužských i ženských hlasů a je možnost výběru modelování prozodie–pomocí neuronových sítí, přímých pravidel, či lineární predikce. Program je zaměřen na paměťovou a časovou efektivitu zpracování a nárok na paměť může být pouze v jednotkách MB [16].

7. Návrh a realizace systému pro syntézu řeči

Úplný systém TTS pro syntézu řeči by měl být schopen převést jakýkoliv vstupní text v českém jazyce na uměle tvořený řečový signál. Zároveň by měl dovolit uživateli modifikovat určité parametry řeči, jako například její trvání, výšku hlasu nebo její intenzitu. Dobrý systém TTS dokáže modifikovat prozodické vlastnosti řeči takovým způsobem, že dokonale simuluje emoce řečníka v daném kontextu a klade správný důraz na předem stanovené hlásky a slabiky.

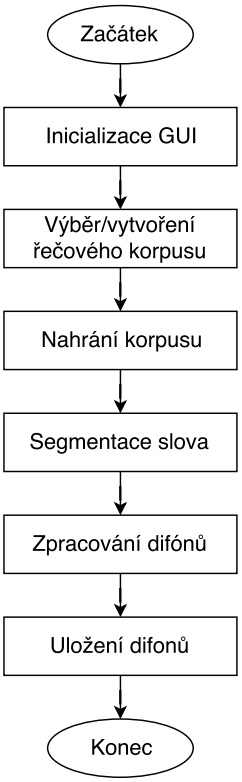
Je důležité zvolit správné řečové jednotky pro oblast použití systému. Pro obecné TTS systémy se zejména používají menší řečové jednotky jako např. difóny nebo trifóny. Použití větších jednotek jako např. slov nebo vět je sice pro lidské ucho přirozenější a je menší problém s řetězením těchto jednotek, je ale téměř nemožné zachytit všechny možné variace jednotlivých slov a požadavek na paměť by byl neúnosný. Inventář difonů by teoreticky měl zahrnovat naprosto všechny kombinace samohlásek a souhlásek. Prakticky je výskyt některých kombinací v daném jazyce vysoce nepravděpodobný (např. 'gf'), je ale nutné se na takový případ připravit (v tomto systému kombinací polofonů pro modelování chybějícího difónu). Řečový korpus bude vytvořen jako součástí práce a to pro dva řečníky (muž a žena), poté analyzován a rozdělen na inventáře řečových jednotek. Součástí programového řešení je tedy i program na nahrávání, analýzu a vkládání jednotek do zvoleného inventáře.

7.1 Nahrání a analýza řečového korpusu

Syntéza řeči na základě techniky lineární predikce závisí na provedené analýze předem nahraného řečového korpusu. Pro jeho nahrání je vhodné vytvořit samostatný program mimo řešení TTS systému pro syntézu řeči. Grafické prostředí tohoto programu je zobrazeno na obrázku 7.11. Vstupní zvukový signál je vzorkován kmitočtem $F_{vz} = 8$ kHz a je vykreslován v časové oblasti, doplněný spektrogramy.

Program také disponuje možností výběru ukládaného inventáře. Uživatel nahraje řečový signál klidným, monotónním hlasem bez emocí dle zobrazeného textu na obrazovce. Takto uživatel pokračuje až do naplnění řečového korpusu. Poté nastává čas analýzy, kde člověk–expert vymezí pomocí kurzorů hranice nahraných hlásek.

Jedná se o vcelku subjektivní proces a je tedy nutná určitá předešlá znalost tvaru obálky hlásek v časové oblasti nebo alespoň schopnost vidět rozdíl v jejich přechodu. Programové řešení zadané úlohy bylo tvořené ve vývojovém prostředí MATLAB R2016a.



Obrázek 7.1: Obecný diagram funkce programu analýzy.

Inicializace GUI

Při startu grafického prostředí programu analýzy se zavolá vytvořená funkce *initData*. Obstarává deklarace a definice globálních proměnných, které se využívají dále v programu. Jedná se např. o proměnnou *speechCorpus*, která v sobě uchovává řečový korpus, tedy zvukové soubory předem nahraného slovníku, které se budou následně analyzovat. Resetují se grafické objekty a v osách se povolí mřížka, provede se prvotní nastavení posuvníků a vyplní se grafické objekty typu *listbox*. Následně dle funkce *f_update* aktivuje program, číslování a zobrazí spektrogramy.

Výběr/vytvoření řečového korpusu

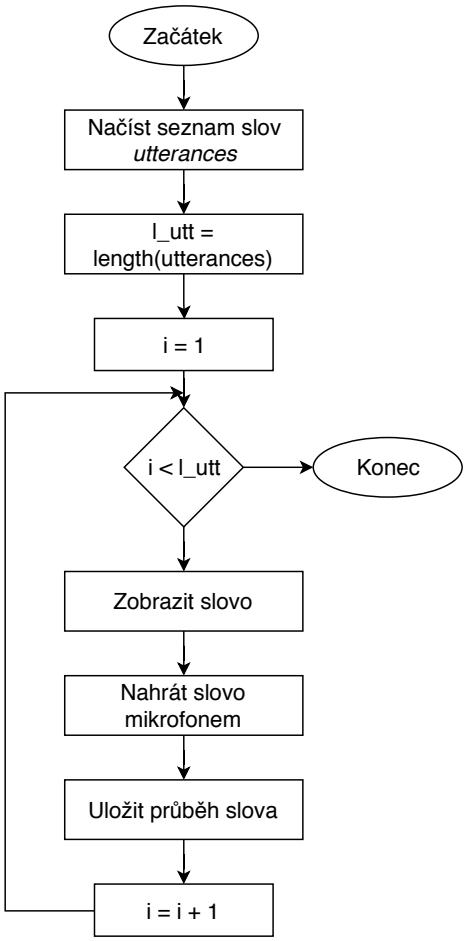
Po startu grafického prostředí má uživatel možnost zvolit existující řečový korpus/slovník, či vytvořit nový. Po výběru souboru se okno obnoví, zobrazí obsah prvního namluveného slova v korpusu a jeho korespondujícími spektrogramy.

Nahrání korpusu

Řečový korpus se nahrává přímo v programu. Uživateli jsou postupně zobrazovány textové příkazy (uložené v proměnné *utterances*), které má uživatel namluvit. Tato pseudo-slova byla vygenerována náhodně, aby obsahovala kombinaci všech hlásek a tedy také reprezentaci všech

difónů bez opakování a redundance. Je samozřejmě obětována jistá přirozenost v porovnání s řečovým korpusem nahraným normální promluvou a souvislými větami. Je nutné zachovat co nejvíce monotonní řeč se stejnou intenzitou. Je nutné dbát na srozumitelnost a tiché prostředí nahrávání.

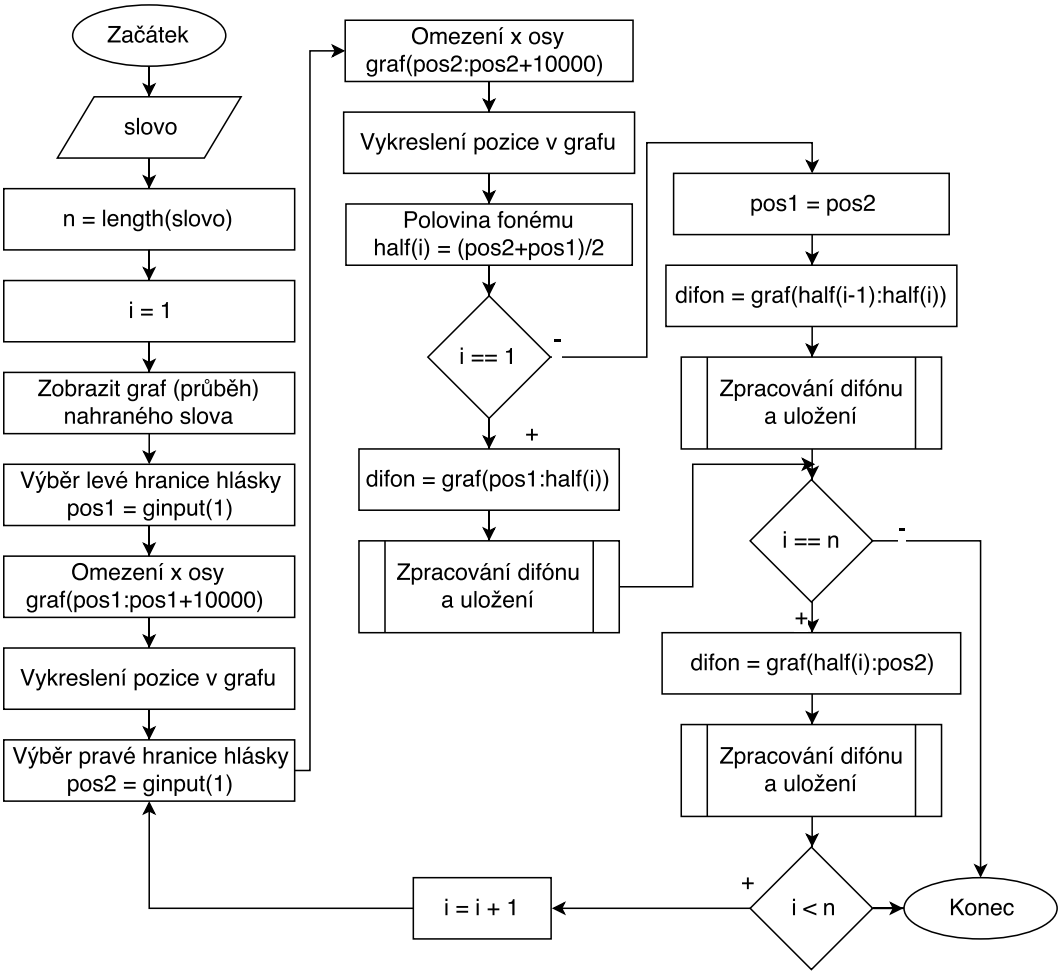
Navržený korpus by obsahoval celkem 453 položek, pokud by byl plně naplněn. Skládá se z 55 střídavých kombinací samohlásek a souhlásek o velikosti 8 písmen a zbytek jako kombinace souhláska-souhláska.
Tento systém byl navržen aby v případě namluvení pouze těchto 55 textových příkazů stále fungoval. Při procesu analýzy se uloží také tzv. *polofony*. Současně se ukládá první polovina první hlásky a druhá polovina poslední hlásky a případně chybějící difóny se pak skládají pomocí těchto doplňujících jednotek.



Obrázek 7.2: Diagram funkce nahrání řečového korpusu.

Segmentace slova

Namluvené slovo se musí rozdělit na jednotlivé řečové jednotky. Uživatel pomocí kurzorů postupně vymezuje hranice fonémů podle spektrogramu a tvaru průběhu. Počet kurzorů je stanovený jako $N + 1$, kde N je počet písmen v zobrazeném slově. Dle kurzorů se spočítají poloviční vzdálenosti mezi fonémy a signál se rozdělí na již zmíněné polofony a hlavně na difóny. Jsou tvořeny druhou polovinou jedné hlásky a první polovinou hlásky sousední. Program posílá nařezané průběhy do funkce *f_analyze2*, která tyto segmenty dále zpracuje.



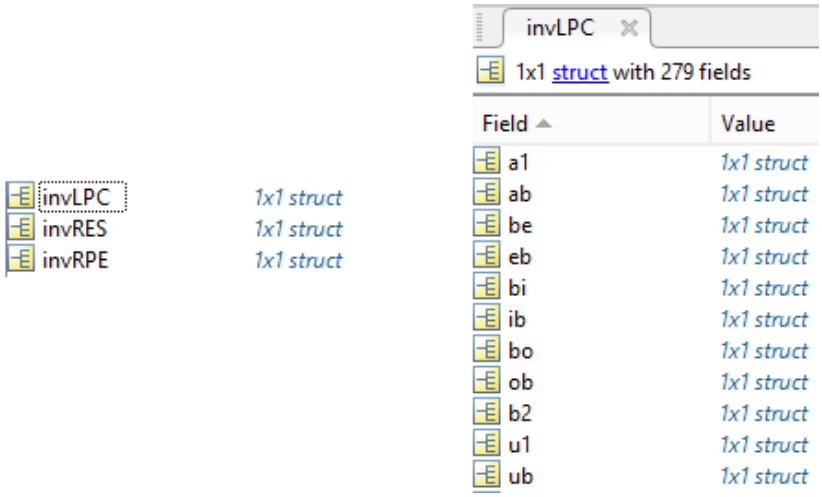
Obrázek 7.3: Diagram funkce segmentace nahraných slov.

Zpracování difónů

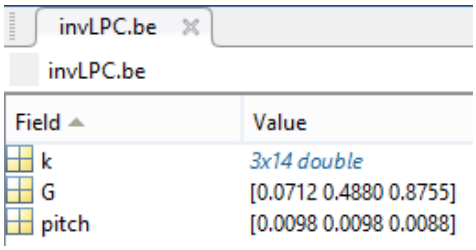
Po segmentaci signálu na řečové jednotky se jednotlivé úseky zpracovávají ve funkci *f_analyze2*. V této metodě se zajistí otevření inventáře a segmentace vstupního signálu na stejné, kvazistacionární úseky o 30 ms. Tyto úseky se dále analyzují ve funkci *f_encode* pro zjištění parametrického popisu zkoumaného segmentu.
Funkce *f_encode* aplikuje na segmenty algoritmus zjištění periody základního tónu řeči T_0 a tedy i F_0. Dále je použit algoritmus Levinson-Durbin pro zjištění autoregresních a reflexních koeficientů navrženého filtru. V algoritmu je zahrnut i výpočet intenzity segmentu.

Uložení difónů

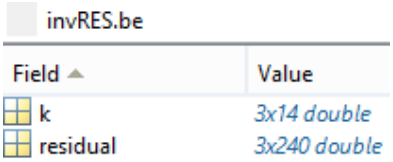
Zjištěné parametry analyzovaných segmentů se ukládají do svých příslušných inventářů *invLPC*, *invRES* a *invRPE* a tyto struktury se zapouzdří do externího souboru **.mat* ve složce *Inventare/*, kde '*' označuje název slovníku.



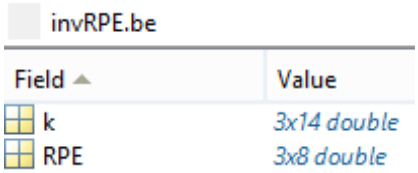
Obrázek 7.4: Ukázka obsahu inventářů řečových jednotek.



Obrázek 7.5: Ukázka obsahu inventáře metody LPC.



Obrázek 7.6: Ukázka obsahu inventáře metody RELP.



Obrázek 7.7: Ukázka obsahu inventáře metody RPE.

7.2 Řečový korpus

Tvorba řečového korpusu byla zjednodušena za pomocí grafického rozhraní, které postupně uživateli dává generované kombinace difonů ve dvou částech. V první části se uživateli nabízí logatomy (střídavé kombinace samohlásek a souhlásek, např. 'abebibob'). Jedná se o jednoduché spojení bez kontextu a smyslu, které se lehce vyslovují a segmentace řečových jednotek je v tomto případě poměrně jednoduchá. V druhé části se vyskytují spojení souhláska-souhláska, které bez párování se samohláskami zní velmi nepřirozeně. Z tohoto důvodu je na začátek a konec těchto spojení souhlásek přidána samohláska, do zpracování se ale nezahrnují. Veškeré zvukové nahrávky se uchovávají v externích souborech ve složce 'Korpusy', která se nachází v kořenovém adresáři programu, s názvem 'XXX.mat', kde 'XXX' je většinou jméno nahrávaného subjektu.

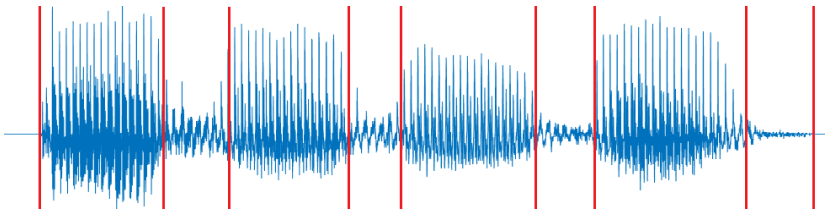
7.3 Inventář řečových jednotek

Inventář je veden ve formátu struktury uložené v externím souboru ve složce 'Korpusy'. Každá řečová jednotka má ve zmíněné struktuře vlastní pole, pojmenované dle jejich kontextu. Jednotky na přechodu dvou hlásek jsou pojmenovány jako kombinace levé a pravé hlásky, jejíž přechod reprezentují (např. 'ah'). Jednotky bez levého kontextu (polofony) se vyjadřují spojením s číslicí '1' a jednotky bez pravého kontextu se spojují s číslicí '2' (např. a1, a2).

Každý difón je rozdělený na určitý počet N překrývaných segmentů, které jsou popsány pomocí N vektorů LPC koeficientů a o velikosti řádu filtru (14), dále pomocí hodnot intenzit G (gain) a period základního tónu (T_0 = pitch), používané jako perioda sledu impulsů při zpětné rekonstrukci znělých částí signálu. Je-li hodnota T_0 nulová, daný segment se klasifikuje jako neznělý a pro budící signál se použije bílý šum.

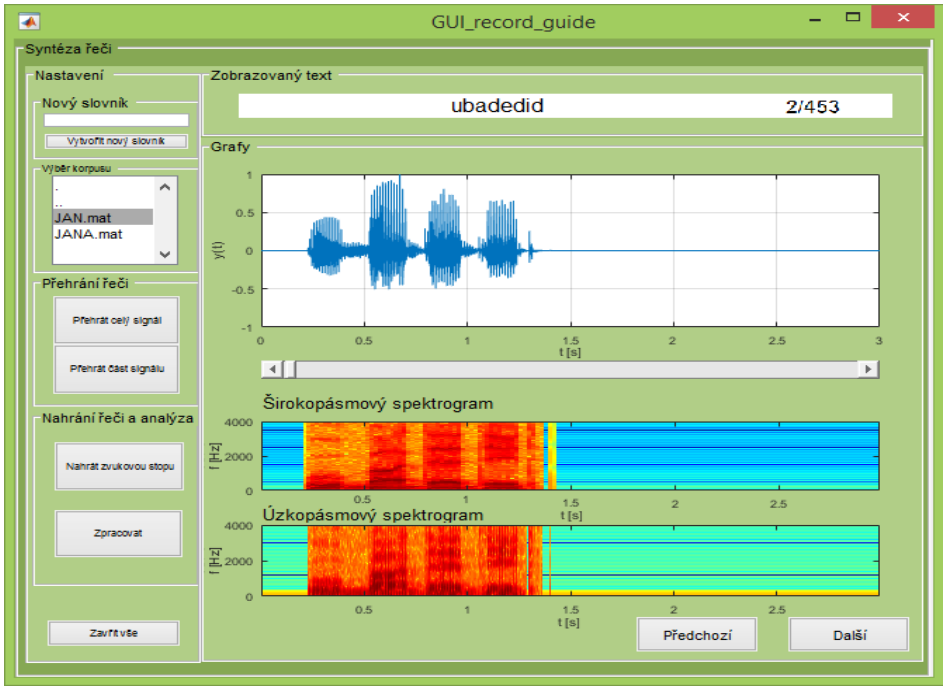
7.4 Zpracování řečových jednotek

Zpracování jednotlivých řečových jednotek probíhá automaticky po určení hranic fonémů uživatelem. Tlačítko *Zpracovat* v grafickém rozhraní postupně zobrazí N+1 kurzorů *ginput*, kde N je počet písmen ve výroku. Uživatel vybere vždy začátek nového fonému. Tento proces vyžaduje určitou úroveň znalosti charakteristik fonémů. Vě většině případů lze hranice celkem jednoduše poznat (např. obrázek 7.8). U některých přechodů je ale tato hranice velmi subjektivní a těžko poznatelná.



Obrázek 7.8: Výběr hranic na logatomu *ahehihoh*.

Zpracování jednotek probíhá kontinuálně při výběru hranic. Na začátku slova se separuje difón bez levého kontextu, zpracuje a uloží do inventáře řečových jednotek. Zároveň se vypočítá horizontální pozice poloviny fonému pro další iteraci. Jelikož difón je řečová jednotka zachycující kontext levého i pravého fonému se v další iteraci pro separaci následující difónu použije polovina předchozího fonému a polovina současného fonému. U poslední iterace se separuje i poslední jednotka s prázdným kontextem. Jako příklad uvedu výraz 'ahoj'. Postupné iterace produkují následující difóny: a1, ah, ho, oj, j2.



Obrázek 7.9: Grafické prostředí programu analýzy *GUI_record_guide*.

7.5 Generování syntetické řeči

Budící signál filtru, který je popsán právě zjištěnými koeficienty je závislý na použité metodě syntézy. V tomto programovém řešení je k dispozici LPC syntéza s jednoduchým modelem buzení puls/šum, dále metoda RELP a nakonec RPE-LPC.

Pro účely LPC syntézy se přenáší reflexní koeficienty filtru k, hodnoty intenzity segmentu G a hodnoty periody základního tónu T_0, pojmenován jako *pitch*. Koeficientů k je stejně jako řádu filtru (14). Koeficienty G a *pitch* platí pro celý segment, takže jich najdeme pouze tolik, kolik je segmentů v řečové jednotce.

Reziduální signál se programově získá jednoduše. Nejprve se vytvoří filtr inverzní k právě zjištěným koeficientům. Jelikož se v tomto řešení využívá reflexních koeficientů PARCOR, bylo jednodušší nejdříve zjistit přenosovou funkci filtru a vytvořit inverzní filtr obrácením čitatele a jmenovatele přenosové funkce. Reziduální buzení získáme, pokud do navrženého filtru pustíme původní analyzovaný signál.

Získání signálu pro metodu RPE-LPC je vcelku jednoduché. V poměru komprese 8:1 je reziduální signál o 240 vzorcích reprezentovaný pouze 30 vzorky na stejně vzdálených pozicích. Ostatní hodnoty mimo pozice vzorků jsou nastaveny na nulu.

Po stisknutí tlačítka *Generace řeči* v grafickém prostředí se provedou následující operace:

Nejprve se pomocí definované funkce *resetAxes(handles)* připraví všechny grafické objekty axes (osy), které později slouží pro vykreslování průběhů. Smaže se jejich obsah pro případ, že v nich zůstaly z předchozí iterace grafické objekty a povolí u nich jemnou mřížku. Dále se deklarují a definují důležité proměnné, jako např. vektor výstupního syntetického signálu *output*, řád filtrů, vzorkovací frekvence F_{vz} a další. Do proměnné *input* se načte řetězec znaků vstupního textu a ošetří se vstup tak, že povolí pouze vstupní znaky definované v proměnné *allowedChar*. Text se dále zpracuje ve funkci *inputProcess(input)*. Tato funkce řeší přepisování vstupních vět, které jsou psané přirozeně s diakritikou na řetězec znaků používaných TTS systémem. V první části přepisuje číselné hodnoty na jejich fonetické podoby (např. číslici 4 na 'tStiRi') a dále se zabývá přepisováním písmen z diakritickými znaménky (např. "má kočka mě olízla" na "ma: kotSka mNe oli:zla"). Znak ':' se bere jako zvláštní označení a to v tom, že poukazuje na místo prodloužení hlásky (např. ó se v textu přepíše jako o:). Nakonec se načtou inventáře řečových jednotek zvolené osoby dle grafického objektu *listboxInv*.

7.6 Řetězení parametrů řečových jednotek

Program obsahuje dvě velké smyčky. Smyčka řetězení jednotek projíždí všechny znaky vstupního vektoru *input*. Pro každý znak v této smyčce se ověří několik podmínek a na jejich základě se vybírají řečové jednotky z vybraného inventáře. Ty se řetězí do jedné společné struktury *data* pomocí funkcí *addUnit* a *addUnit2*. Ta předchozí se pokusí najít požadovaný difón v inventáři jednotek a předá jej do druhé funkce. Pokud se v inventáři nenachází, zhotoví se syntetický difón ze dvou polofonů (difón 'gh' se zhotoví řetězením polofonu 'g2' a 'h1'). Funkce *addUnit2* vezme načtená data řečové jednotky, zřetězí parametry (např. koeficienty filtru, periodu základního tónu a zesílení segmentu) a pokusí se vyhladit spektra mezi sousedícími jednotkami pomocí rovnic 5.19.

Pro tento algoritmus byl definován vektor, který definuje co se považuje za speciální znak (většinou znaky vymezující mezery v textu) a je vytvořená proměnná *start*, která označuje začátek slova. Uvažuje se podmínka 'A' (proměnná start==0), podmínka 'B' (současný znak se považuje jako speciální) a podmínka 'C' (další znak v iteraci se považuje jako speciální).

Výstupní akce se dá rozdělit v podstatě na 5 možných výsledků. Logická kombinace podmínek ($A \wedge \overline{B} \wedge \overline{C}$) provede první akci (přidá řečovou jednotku na začátku slova–polofon a nastaví proměnnou start=0). Kombinace ($\overline{A} \wedge \overline{B} \wedge \overline{C}$) provede druhou akci (přidá obyčejný difón). Kombinace ($\overline{A} \wedge \overline{B} \wedge C$) provede třetí akci (přidá difón, zakončí jednotku polofonem a nastaví proměnnou start=1). Poslední dvě kombinace ($A \wedge B \wedge C$) a ($A \wedge B \wedge \overline{C}$) přidají mezeru o určité velikosti pomocí funkce *addSpace()*. Jedná se o podobnou funkci jako *addUnit*, jenže řetězené parametry jsou náhodně generovány jako velmi slabý šum. Důvod použití šumu místo pouhého nahrazení nulami je přirozenější zobrazení tichých regionů ve spektrogramech. Použitím nul vzniká ve spektrogramu prázdné místo a velmi ovlivní jeho škálu. Pro lepší pochopení tohoto algoritmu je v tabulce 7.1 zobrazen názorný případ vstupního řetězce a provedení výsledných akcí.

Tabulka 7.1: Ukázka algoritmu výběru řečových jednotek.

ii	input(ii)	A	B	C	Provedená akce
1	a	1	0	0	addUnit(‘a1’), (start=0)
2	h	0	0	0	addUnit(‘ah’)
3	o	0	0	0	addUnit(‘ho’)
4	j	0	0	1	addUnit(‘oj’), addUnit(‘j2’), (start=1)
5	,	1	1	1	addSpace(2)
6		1	1	0	addSpace(1)
7	j	1	0	0	addUnit(‘j1’), (start=0)
8	a	0	0	0	addUnit(‘ja’)
9	k	0	0	1	addUnit(‘ak’), addUnit(‘k2’), (start=1)
10		1	1	0	addSpace(1)
11	s	1	0	0	addUnit(‘s1’), (start=0)
12	e	0	0	1	addUnit(‘se’), addUnit(‘e2’), (start=1)
13		1	1	0	addSpace(1)
14	m	1	0	0	addUnit(‘m1’), (start=0)
15	á	0	0	0	addUnit(‘ma’)
16	š	0	0	1	addUnit(‘aS’), addUnit(‘S2’), (start = 1)
17	.	0	1	0	addSpace(2)

7.7 Tvorba syntetických segmentů

Všechny parametry řetězených řečových jednotek již máme z předchozího kroku uloženy ve struktuře *data*. Pro syntézu jsou použity dvě vnořené smyčky. První projíždí všechny uložené řečové jednotky, každá se svým vlastním indexem *it*. Každá jednotka může být reprezentovaná X počtem segmentů. Druhá smyčka projíždí všechny segmenty současné řečové jednotky.

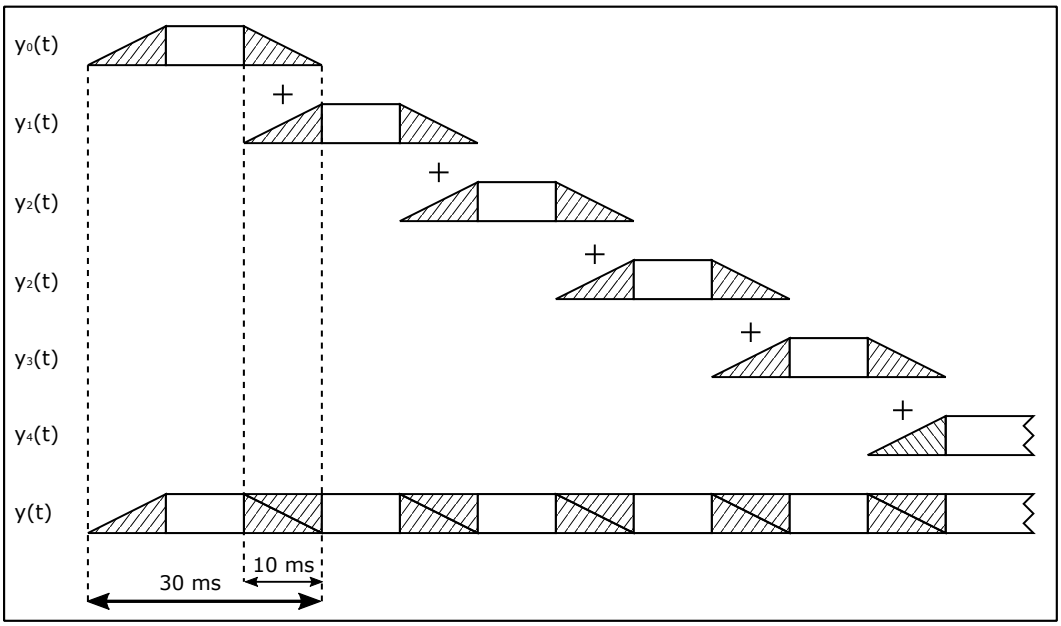
V grafickém prostředí je při syntéze pomocí jednoduchého modelu LPC možné zajistit trvalé buzení pulsy nebo trvalé buzení bílým šumem. Za tohoto případu je tedy na začátku druhé smyčky změněn parametr *pitch* periody základního tónu na konstantní hodnotu pro znělou klasifikaci, či na nulovou hodnotu pro neznělou klasifikaci. Dále je řešeno prodloužení označených hlásek symbolem ‘:’. Jejich pozice je označena v proměnné *indLong* a prodlužuje velikost vytvářených umělých segmentů daného difónu. Jelikož difón obsahuje informace dvou různých hlásek, prodlužujeme pouze druhou polovinu prvního difónu a první polovinu druhého difónu.

Budící signál pro jednoduchý model LPC je tvořen na základě parametru *pitch*, tedy periody základního tónu řeči. V případě nulové hodnoty byl segment v sekci analýzy klasifikován jako neznělý a budící signál filtru je tvořen bílým šumem s variancí 1 o velikosti segmentu.

Hodnota nad nulu znamená klasifikace segmentu jako znělý a budící signál tvoří sled pulsů o velikosti 1 a periodou parametru *pitch*. Tento zdrojový signál se poté násobí intenzitou segmentu, uloženou v proměnné G. Pro syntézu byl použit filtr s křížovou strukturou, tvořený koeficienty odrazu (PARCOR), získaných přímo z Levinson-Durbin algoritmu 5.10. Umělý segment řeči se získá průchodem budícího signálu filtrem s křížovou strukturou. Přenosová funkce tohoto filtru se zachová pro pozdější informativní zobrazení v grafickém prostředí.

7.8 Rekonstrukce výsledného signálu

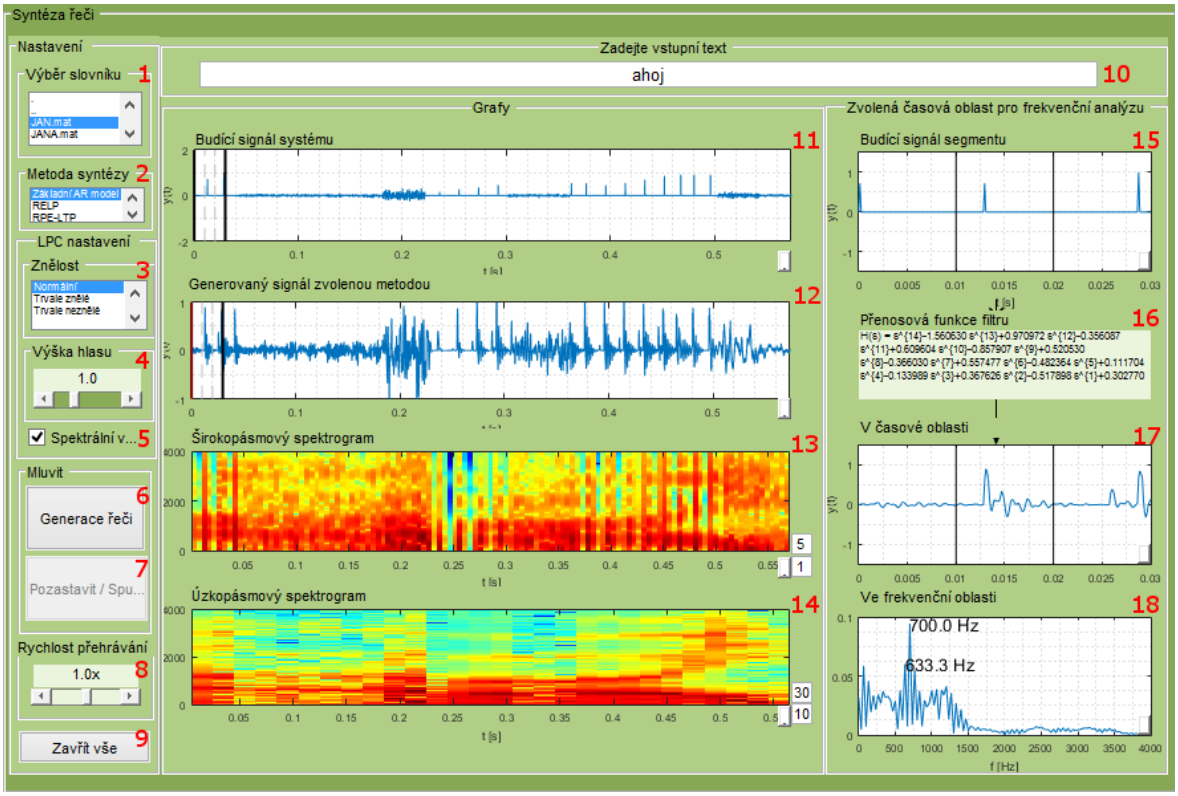
Rekonstrukce výsledného syntetického signálu se provádí průběžně po vytvoření jednotlivých umělých segmentů. Právě tvořený segment řeči se skládá přes předchozí segmenty tak, aby byla určitá část překryta. Oblast překrytí je standardně 80 vzorků (10 ms), avšak tato hodnota se může měnit v závislosti na zvolené přehrávací rychlosti. Vždy je ale poměrově stanovena jako třetina délky segmentu. Překrývaná část signálu se extrahuje a jejich hodnoty se zprůměrují. Výsledný signál se pak získá skládáním překrývané oblasti a zbytku tvořeného segmentu. Princip lze vidět na obrázku 7.10.



Obrázek 7.10: Rekonstrukce řečového signálu.

7.9 Grafické prostředí TTS systému

V této části práce budou popsány jednotlivé části hlavního grafického prostředí pro syntézu řeči. Okno programu lze vidět na obrázku 7.11. Grafické objekty jsou zvýrazněny a označeny červenými čísly. V tabulce 7.2 lze vyčíst jejich účel v programu.



Obrázek 7.11: Grafické prostředí realizovaného TTS systému.

Tabulka 7.2: Popis grafických objektů v programu.

Číslo objektu	**Typ objektu**	**Popis funkce**
1	listbox	Výběr inventáře řečových jednotek (Muž, žena).
2	listbox	Výběr použité metody syntézy řeči. Je možná LPC syntéza s jednoduchým modelem buzení puls/šum, metoda RELP nebo metoda RPE-LPC.
3	listbox	Při výběru LPC syntézy lze programu volitelně přikázat, aby byly všechny tvořené segmenty znělé, či neznělé.
4	slider	Uživatel má na výběr ovlivňit analyzovanou výšku hlasu. Rozmezí je nastaveno jako 0.25–3x analyzované periody.
5	checkbox	Zapíná/Vypíná vyhlazení spektrálních nespojitostí mezi řetězenými řečovými jednotkami.
6	pushbutton	Hlavní tlačítko, které generuje syntetický signál.
7	pushbutton	Po vygenerování řeči lze pomocí tohoto tlačítka pozastavit, či znovu spustit syntetický signál.
8	slider	Možnost změny rychlosti přehrávání. Rozmezí posuvníku je dáno od 0.5–1.5x, tedy polovina původní rychlosti až dvojnásobná. U LPC metody se provádí změnou délky syntetického segmentu a působí tedy přirozeně. U ostatních metod je tomu docíleno pouze změnou přehrávací frekvence.
9	pushbutton	Tlačítko zavře všechny grafické objekty a ukončí program.
10	edit	Textové pole, které slouží pro zadání vstupního textu, který se má následně převést na řečový signál.
11	axes	Osa, do které se vykresluje budící signál systému.
12	axes	Osa, do které se vykresluje hlavní, synteticky vytvořený signál, který vznikne průchodem budícího signálu vhodně tvořenými filtry.
13	axes	Širokopásmový spektrogram.
14	axes	Úzkopásmový spektrogram.
15	axes	Osa, která zobrazuje budící signál zvoleného segmentu o délce 30 ms.
16	axes	Zobrazuje přenosovou funkci filtru zvoleného segmentu.
17	axes	Osa, která zobrazuje výsledný syntetický segment. Získá se průchodem budícího signálu z objektu 15 skrze filtr, jehož přenosová funkce je v objektu 16.
18	axes	Osa, která zobrazuje výsledný syntetický řečový segment, převedený do frekvenční oblasti pomocí FFT. Informativně je doplněný popisem dvou nejvyšších vrcholků.

7.10 Vyhodnocení výsledků

V rámci syntézy libovolné promluvy TTS systémů není objektivní zhodnocení kvality umělé řeči v současné době možné a musíme se odkazovat na subjektivní poslechové testy [11]. Zaměřujeme se přitom na srozumitelnost a přirozenost.

Jedním z nejvíce používaných testů srozumitelnosti řeči je **test modifikací rýmu** (Modified Rhyme Test–MRT). Zaměřuje se především na souhlásky, jelikož správná syntéza souhlásek má velmi velký vliv na celkovou srozumitelnost řeči. Jsou generovány skupiny jednoslabičných slov, které se od sebe liší pouze v první, či koncové souhlásce. Tyto skupiny lze vidět v tabulce 7.3. Posluchač vybere z nabízené skupiny slov právě to slovo, které uslyší. Po přehrání určitého počtu slov je možné provést vyhodnocení. Výhodou tohoto testu je jeho spolehlivost a malý počet posluchačů nutných k vykonání testu. Nevýhodou může být omezený počet rýmujících slov ve skupině a posluchač může své rozhodnutí ovlivnit touto nabídkou [11].

Tabulka 7.3: Skupiny slov pro vyhodnocení srozumitelnosti řeči (test MRT).

pyl	pih	pij	piš	pir	pin
pes	les	mez	bez	děs	rez
rak	tak	sak	jak	lak	pak
lef	les	lem	lep	led	len
byt	lid	kyt	žid	hit	šít
loj	lom	lot	lok	los	lob
kos	bos	los	nos	šos	sos
bál	šál	tál	gál	sál	kál
suk	puk	kuk	luk	muk	fuk
říz	líz	míz	fíz	sís	ciz
bez	lez	fez	rez	pez	jez
buk	puk	suk	fuk	muk	tuk
rus	ruch	rum	rub	ruk	rur
mop	mor	mok	moc	mol	moč
sek	šek	řek	dek	bek	jek

Pro tento účel byl vytvořen program v MATLABu pro jednodušší záznam odpovědí a zajištění randomizace, kvůli dodržení objektivity testu. Posluchači se nabízí náhodně seřazené a vybrané kategorie slov a ten pomocí čísel 1–6 označí to slovo, které v nahrávce slyšel. Program je spojen s již vytvořeným syntetizérem přes *GUI_speak*. Metoda, kterou se slovo syntetizuje je také vybraná náhodně, ale je zajištěno, aby byla každá metoda reprezentovaná ve stejném počtu. Data se vyhodnocují dle počtu správně vybraných slov. Celkově bylo použito patnáct skupin po šesti slov, které byly zhodnoceny 14 lidmi.

Tabulka 7.4: Vyhodnocení testu MRT.

Metoda	**Bodové výsledky (0–210)**
Model puls/šum	160
RELP	170
RPE-LPC	159

Z tabulky 7.4 lze vyvodit, že mezi první a třetí metodou syntézy není velký rozdíl ve srozumitelnosti na testovaných skupinách slov a jsou porovnatelné. Předvídatelně, nejvíce správně zvolených slov bylo při použití metody RELP. Při zaznamenávání dat bylo zjištěno, že posluchačům dává největší problém rozpoznání ploziv (p, b, t, d, k,...), naopak nejsnazší byly např. slova s určitými frikativy a afrikáty (š, s, c, č, r,...).

Velmi rozšířeným testem přirozenosti řeči je **test MOS** (Mean Opinion Score), který byl definovaný Mezinárodní telekomunikační unií a posuzuje řeč na základě hodnocení posluchače v rozmezí od 1–5. V tabulce 7.5 lze vidět, že se hodnotí buď kvalita řeči, či námaha, která je potřebná k porozumění dané věty. Výsledky od všech posluchačů se zprůměrují do výsledného skóre metody/syntetizéru. Testované věty zahrnují 'ahoj, jak se máš', 'mám se dobře', 'tohle je umělá řeč', 'k vánocím si přeji auto', 'mám rád kočky', 'kočky nemají rády mě', 'třistatřiatřicet stříbrných stříkaček', 'včera jsem dostal mobilní telefon', 'pozdě bycha honit', 'než řekneš švec', 'kozel je populární české pivo', 'chleba se šunkou', 'mám k prodeji starý počítač', 'potřebuji vyčistit komín na domě' a 'hovězí a kuřecí maso'.

Pro tento test byl také vytvořen pomocný program, který provede randomizaci výběru vět i použité metody syntézy a poskytuje jednoduché zadávání i ukládání dat. Vyhodnocení v tomto případě je ve formě průměrné známky hodnocení posluchače. Systém byl testován celkem 14 posluchači.

Tabulka 7.5: Tabulka hodnocení pro testy MOS (převzato z [11]).

Hodnocení	**Kvalita řeči**	**Vynaložená námaha při poslechu**
1	špatná	zcela neodpovídající
2	horší	veliká
3	průměrná	průměrná
4	dobrá	bez větší námahy
5	výborná	bez jakékoliv námahy

Tabulka 7.6: Vyhodnocení testu MOS.

Metoda	**Průměrná známka hodnocení (0–5)**
Model puls/šum	3,10
RELP	4,09
RPE-LPC	2,92

Z výsledků tabulky 7.6 lze vidět, že syntéza pomocí techniky RELP měla zdaleka nejvyšší hodnocení kvality řeči. Z principu byl tento výsledek očekáván, jelikož budící signál filtrů segmentů není zjednodušován a měl by se tedy jevit nejkvalitnějším projevem.

Dále můžeme porovnat syntetické signály dle jejich požadavků na paměť a přenos parametrů. V základu MATLAB ukládá všechny numerické hodnoty jako double-precision floating-point, který zabírá 8 bytů místa (64 bitů).

U základního modelu LPC s budícím signálem impulsy/šum je na každý segment řeči použitý filtr 14. řádu, je tedy popsaný 14 koeficienty k, tedy $14 \cdot 64 = 896$ bity. Dále je popsán hodnotou intenzity segmentu G a periodou základního hlasivkového tónu v proměnné *pitch*. V tomto formátu by tedy koeficienty, popisující jeden segment zabíraly $15 \cdot 64 = 1024$ bitů. Na jednu sekundu výsledného řečového signálu je zapotřebí vytvořit a zřetězit celkově $\frac{1}{0.02} = 50$ segmentů a tedy pro jednu sekundu tvořeného signálu pomocí jednoduchého modelu puls/šum je nutné převést 51200 bitů dat, nebo-li 6,4 kB/s.

Model RELP nepopisuje budící signál filtru parametricky, ale přenáší se celý. Bez využití komprimačních technik je tedy popsaný 240 vzorky hodnot. Každý vzorek zabírá v základní definici již zmíněných 64 bitů, signál tedy bude zabírat $240 \cdot 64 = 15360$ bitů na buzení. Filtr je popsaný standardně pomocí k koeficientů, zabírá tedy 896 bitů při datovém typu double. Celkově tedy při 1 sekundě výsledného signálu je nutno přenést $(15360 + 896) \cdot 50 = 812800$ bitů, nebo-li 101,6 kB/s.

Technika RPE-LPC navazuje na RELP v tom, že budící signál je reprezentován pouze pomocí pár impulsů. V poměru komprese 8:1 je reziduální signál o typické velikosti 240 vzorků popsaný $240/8 = 30$ pulsy. Zabírá tedy $30 \cdot 64 = 1920$ bitů a pro 1 s syntetické řeči je třeba přenést $(1920 + 896) \cdot 50 = 140800$ bitů, nebo-li 17,6 kB/s.

Při specifikaci datového typu na *single*, který je popsaný pouze 4 byty, nebo-li 32 bity se přenosová rychlost zkrátí o polovinu.

Tabulka 7.7: Potřebný přenos dat pro syntézu řeči.

Datový typ	**Sled pulsů/šum**	**RELP**	**RPE-LPC**
Double	6,4 kB/s	101,6 kB/s	17,6 kB/s
Single	3,2 kB/s	50,8 kB/s	8,8 kB/s

8. Závěr

Hlavním cílem této práce bylo navrhnout a realizovat systém pro konkatenační syntézu řeči s využitím techniky lineární predikce jako model hlasového traktu. Celé programové řešení bylo vytvořeno čistě v prostředí MATLAB R2016a.

V rámci práce byly pro syntézu řeči implementovány tři metody, které se liší svým budícím signálem. Jedná se o metodu se zjednodušeným buzením puls/šum, dále metodu RELP, která používá signál chyby predikce jako buzení filtru a nakonec RPE-LPC, která aproximuje signál chyby predikce jako sled rovnoměrně zvolených impulsů. Pro jednodušší interakci s programem bylo vytvořeno samostatné grafické prostředí pro analýzu i syntézu řeči.

Pro posouzení srozumitelnosti a kvality realizovaného syntetizéru a jeho jednotlivých metod (šum/puls, RELP, RPE-LPC) byl systém vyhodnocen poslechovými testy MRT a MOS. Tyto testy byly provedeny u celkem 14 posluchačů a jejich výsledky byly zhodnoceny v tabulkách 7.4 a 7.6. U testu MRT má posluchač za úkol vybrat přehrané slovo ze skupiny jednoslabičných slov, které se liší pouze v první, či poslední hlásce. Vyhodnocení probíhá dle počtu správně zvolených slov. U testu MOS posluchač hodnotí kvalitu řeči subjektivně ze škály hodnot 1–5 (5 je nejkvalitnější řeč). Výsledky všech posluchačů se zprůměrují do celkového skóre metody.

Z výsledků testu MRT, která se zaměřuje na správnou srozumitelnost souhlásek, dosáhla metoda se zjednodušeným buzením puls/šum a metoda RPE-LPC velmi podobným výsledkům (160/210 a 159/210). Podobného závěru můžeme dospět i z výsledků testů MOS, kde metody puls/šum a RPE-LPC dosáhly téměř stejného průměrného hodnocení (3,10 a 2,92). Dle očekávání dosáhla prokazatelně nejlepších výsledků u obou testů metoda RELP, která používá pro buzení filtrů tzv. reziduum (signál chyby predikce). V testu MRT posluchači zvolili správně 170 z 210 nabízených slov a v testu MOS získala průměrného ohodnocení 4,09 z 5.

Dosažená kvalita syntézy velmi závisí na použité metodě i velikosti inventáře řečových jednotek. Velký vliv má i to, že řečové jednotky jsou obsaženy v inventářích pouze jednou. Kvalitnější syntetizéry typicky analyzují řeč s přirozeným projevem a zahrnují vícero reprezentací každé řečové jednotky v různých kontextech. Poté se dynamicky vybírají nejvhodnější zastoupení dle potřeby. Pro vylepšení práce by bylo vhodné implementovat techniku zvanou TD-PSOLA, která nabízí možnost prozodických modifikací u metod RELP a RPE-LPC. Doporučoval bych také experimentaci s algoritmy automatické segmentace řeči (např. pomocí HMM). Program by se také dal s menšími úpravami převést do jazyka C a C++ pomocí MATLAB Coder.

Literatura

[1] Modelování lidského hlasu [online]. Dostupné z: <https://vesmir.cz/cz/casopis/archiv-casopisu/2008/cislo-12/modelovani-lidskeho-hlasu.html>, [cit. 2017-12-10].

[2] Pitch Detection Methods Review [online]. Dostupné z: <https://ccrma.stanford.edu/~pdelac/154/m154paper.htm>, [cit. 2017-12-15].

[3] Performance Evaluation of Pitch Detection Algorithms [online]. Dostupné z: <http://access.feld.cvut.cz/view.php?cisloclanku=2009060001>, [cit. 2017-12-18].

[4] Dutoit, T.: *An Introduction to Text-to-Speech Synthesis*. Text, Speech and Language Technology, Springer Netherlands, 1997, ISBN 9780792344988.
URL <https://books.google.cz/books?id=xjXU3FujWbwC>

[5] Furui, S.: *Digital Speech Processing: Synthesis, and Recognition, Second Edition,*. Signal Processing and Communications, Taylor & Francis, 2000, ISBN 9780824704520.
URL <https://books.google.cz/books?id=AypAngEACAAJ>

[6] Gold, B.; Morgan, N.; Ellis, D.: *Speech and Audio Signal Processing: Processing and Perception of Speech and Music*. Wiley, 2011, ISBN 9781118142899.
URL <https://books.google.cz/books?id=p0mNAF8gMmUC>

[7] Hess, W.: *Pitch determination of speech signals: algorithms and devices*. Springer series in information sciences, Springer-Verlag, 1983, ISBN 9783540119333.
URL <https://books.google.cz/books?id=PWlTAAAAMAAJ>

[8] Huang, X.; Acero, A.; Hon, H.: *Spoken Language Processing: A Guide to Theory, Algorithm, and System Development*. Prentice Hall PTR, 2001, ISBN 9780130226167.
URL <https://books.google.cz/books?id=reZQAAAAMAAJ>

[9] Noll, A. M.: Cepstrum Pitch Determination. *The Journal of the Acoustical Society of America*, rocnik 41, c. 2, 1967: s. 293–309, doi:10.1121/1.1910339.

[10] Psutka, J.: *Komunikace s počítačem mluvenou řečí*. Academia, 1995, ISBN 9788020002037.
URL <https://books.google.cz/books?id=V7inAAAACAAJ>

[11] Psutka, J.; Müller, L.; Matoušek, J.; aj.: *Mluvíme s počítačem česky*. Česká matice technická, Academia, 2006, ISBN 9788020013095.
URL <https://books.google.cz/books?id=3XyyAAAACAAJ>

[12] Sigmund, M.: *Voice Recognition by Computer*. Tectum-Verlag, 2003, ISBN 9783828884922.
URL <https://books.google.cz/books?id=B9VuCBBYzJ4C>

[13] SpeechTech Text-to-speech [online]. Dostupné z: <http://www.speechtech.cz/cz/produkty/speechtech-tts-synteza-reci>, [cit. 2018-04-26].

[14] KobaSpeech - hlasový syntetizér [online]. Dostupné z: <http://www.spektra.eu/cs/zrakove-vady/programy/syntezatory/kobaspeech>, [cit. 2018-04-26].

[15] Acapela Infovox 4 (Eliška) - hlasový syntetizér [online]. Dostupné z: <http://www.spektra.eu/cs/zrakove-vady/programy/syntezatory/acapela>, [cit. 2018-04-26].

[16] Epos - A Free Text to Speech Synthesis System [online]. Dostupné z: <http://epos.ufe.cz/>, [cit. 2018-04-26].

Seznam příloh

K diplomové práci je přiloženo CD s touto adresářovou strukturou:

- `application`
  - zdrojové kódy programů (MATLAB)
  - `Inventare/` — databáze řečových jednotek
  - `Korpusy/` — nahrané úseky řeči
- `pdf`
  - text diplomové práce (PDF soubor)
- `tests`
  - hodnoty poslechových testů MRT a MOS (excel soubor)